SIDE-CHANNEL ANALYSIS OF SUBSCRIBER IDENTITY MODULES

THESIS

John Andrew Hearle

AFIT-ENG-13-J-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-13-J-03

SIDE-CHANNEL ANALYSIS OF SUBSCRIBER IDENTITY MODULES

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science

John Andrew Hearle, B.S.

June 2013

AFIT-ENG-13-J-03

SIDE-CHANNEL ANALYSIS OF SUBSCRIBER IDENTITY MODULES

John Andrew Hearle, B.S.

Approved:

Rusty O. Baldwin, PhD (Chairman)	4 Jun 13 Date
Barry E. Mullins, PhD (Member)	4 Jun 13 Date
Michael A. Temple, PhD (Member)	4 Jun 2013 Date

AFIT-ENG-13-J-03

Abstract

Cellular phones contain a wealth of data useful to forensic investigators. Much of this data is stored within the hardware of the phone itself, but other data is often contained in a small card-like device called a subscriber identity module (SIM). SIMs are often locked with a PIN code that prevents access to the data unless the correct password is entered. If an invalid PIN is entered several times, the card locks and may even destroy its stored data. This presents a challenge to the retrieval of forensic data from the SIM when the PIN is unknown.

The field of side-channel analysis (SCA) collects, identifies, and processes information leaked via inadvertent channels. One promising side-channel leakage is that of electromagnetic (EM) emanations; by carefully monitoring the SIM's emissions, it may be possible to identify specific operations being performed by the internal microprocessor, and thus, determine the correct PIN to unlock the card.

This thesis uses EM SCA techniques to attempt to discover the SIM card's PIN. The tested SIM is subjected to simple and differential electromagnetic analysis. No clear data dependency or correlation is apparent. The SIM does reveal information pertaining to its validation routine, but the value of the card's stored PIN does not appear to have an effect on EM leakage, and thus the PIN cannot be determined. The two factors contributing to this result are the black-box nature of PIN validation and the hardware and software SCA countermeasures. Further experimentation on SIMs with known operational characteristics could be used to reveal data dependencies and countermeasure effects, determining the viability of future SCA attacks on these devices.

Acknowledgments

I would like to thank my thesis advisor, Dr. Rusty Baldwin, for his patience in guiding me through this research process. I would also like to thank Mr. Paul Simon and Mr. Pranav Patel for their assistance in the lab.

In addition, I would like to thank my fiancée for her daily support, patience, and for helping me edit many late-night revisions of this thesis.

Finally, I would like to thank God for giving me the ability to conduct this research, and for blessing my life with the great people listed above.

John Andrew Hearle

Table of Contents

Page

Page

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
2G	second-generation
3G	third-generation
ADF	application dedicated file
ADN	Abbreviated Dialing Numbers
AES	Advanced Encryption Standard
APDU	Application Protocol Data Unit
API	application programming interface
ATR	answer to reset
BNC	Bayonet Neill-Concelman
COTS	commercial off-the-shelf
CPU	central processing unit
DEMA	Differential Electromagnetic Analysis
DES	Data Encryption Standard
DF	dedicated file
DPA	Differential Power Analysis
EEPROM	electrically erasable programmable read-only memory
EF	elementary file
EMA	electromagnetic analysis
EM	electromagnetic
FFT	fast Fourier transform
FIB	focused ion beam
FPGA	field-programmable gate array
GSM	Global System for Mobile Communications

Acronym	Definition
ICCID	integrated circuit card identifier 5
IMSI	international mobile subscriber identity 5
JTAG	Joint Test Action Group 41
JVM	Java virtual machine 8
LNA	low-noise amplifier 21
LND	Last Numbers Dialed 5
LOCI	Location Information 6
LTE	Long-Term Evolution 13
MF	master file 9
NIST	National Institute of Standards and Technology 5
OS	operating system 8
PCB	printed circuit board 37
PC	personal computer 8
PIN	Personal Identification Number 1
PLL	phase-locked loop 8
PSD	power spectral density 47
PUK	Pin Unlock Key 4
RAM	random-access memory 6
RNG	random number generator 6
ROM	read-only memory 6
RSA	Rivest-Shamir-Adleman 8
SCA	Side-Channel Analysis 23
SEMA	Simple Electromagnetic Analysis 22
SIM	Subscriber Identity Module 2
SMS	Short Message Service 6

Acronym	Definition
SPA	Simple Power Analysis 19
SUT	system under test 32
SoC	system-on-chip 7
UART	universal asynchronous receiver/transmitter 13
UMTS	Universal Mobile Telecommunications System 12
USB	Universal Serial Bus 17
VHDL	VHSIC Hardware Description Language 39
LAC	Location Area Code 6

SIDE-CHANNEL ANALYSIS OF SUBSCRIBER IDENTITY MODULES

I. Introduction

1.1 Motivation

The field of side-channel analysis (SCA) collects, identifies, and processes information unintentionally leaked through inadvertent channels in software, systems, and devices. Examples of common side channels include the completion time for a task or process, the amount of power consumed by a device during an operation, and electromagnetic emanations from a semiconductor device. Each of these side channels is capable of revealing protected information to an attacker, details about the system's operation, or a combination of the two. Leaking a password or encryption key via a side channel can grant an attacker access to the rest of the system through its intended channels.

This research applies side-channel analysis to subscriber identity modules, or SIM cards. A SIM holds a cellular network subscriber's identity, which is used to authenticate the user to the network. In addition, the SIM may contain various types of user data useful to forensic investigators, such as phonebook entries, text messages, and location information. This information is often protected with a Personal Identification Number (PIN) to prevent unauthorized access. Because of the SIM's role in network authorization, considerable research has been directed toward making it a highly secure device. Although the PIN is typically a simple four-digit numeric code, its security is enhanced by a retry counter that locks the card if an incorrect PIN is entered three times, ruling out the possibility for a brute-force attack.

This security measure blocks unauthorized access to the SIM's data through the primary channel, and a SIM card does not have any alternate primary channels available.

As such, an attacker must locate a side channel to extract the PIN in order to obtain the forensic data.

1.2 Goals and Limitations

This research examines the possibility of exploiting PIN leakage through the electromagnetic side channel. A methodology for attacking the PIN via electromagnetic analysis is presented with multiple capture methods to increase the probability of success. Both simple and differential electromagnetic attacks are shown. The ultimate goal is to produce a working attack that recovers the PIN. Barring complete success, other goals include the examination of electromagnetic attacks against modern SIMs and evaluating the effects of countermeasure implementations, as current research against SIM card PINs is rather limited. However, design secrecy of SIM countermeasures prevents a full understanding of which countermeasures to expect and in what form to expect them.

Research limitations include the number of SIMs available, limitations in the collection hardware and interfacing device, and details about the device's internal architecture. This research is not intended as an exhaustive search of all electromagnetic attacks on all smart cards. Only a select few of each are chosen, and the research focuses specifically on Subscriber Identity Module (SIM) cards. Even though much of this research is applicable to other forms of smart cards, they are not considered here.

1.3 Thesis Layout

This document is divided into the following chapters. The second chapter, following this introduction, covers the background of SIM cards and the development of the side channel analysis field. It also presents countermeasures in SIM cards designed to frustrate attempts at side channel analysis attacks. Chapter three outlines a research methodology to execute an electromagnetic analysis attack against the SIM card's PIN. Chapter four records the results of that attack and evaluates its effectiveness. The final chapter points

toward future work opportunities to carry out further evaluation of such attacks against SIM card PINs.

II. Background

2.1 Introduction

This chapter introduces subscriber identity modules (SIMs), side channel analysis, and the interaction between the two. Section 2.2 presents the module itself and a description of the forensic data contained therein. It also discusses the module's architecture and the security features in place to protect the card from unauthorized access and use. Section 2.3 describes the field of side-channel analysis and discusses important historical attacks and their development. Section 2.4 notes previous research where side-channel analysis is applied to smart cards and SIMs. Finally, Section 2.5 describes countermeasures implemented on SIMs to protect against side-channel analysis and what effect these features are expected to have on attacks.

2.2 Subscriber Identity Module (SIM) Overview

2.2.1 Introduction to SIMs.

Subscriber identity modules (SIMs) are small, card-like devices used in mobile phones to identify and authenticate the user to the network. They perform this task based on strong cryptographic algorithms and keys contained within the card. In addition to network-based identity information, the SIM also contains personal data relevant to the user, such as text messages, recently dialed numbers, and location information. Both authentication and personal data is protected by a Personal Identification Number (PIN), a four-digit numeric code. This PIN, known to the subscriber, is protected against brute force attacks by enforcing a three-strike lockout policy. If the PIN is entered incorrectly three times in a row, the SIM locks down and can no longer be unlocked by the PIN. Instead, a Pin Unlock Key (PUK) must be obtained, usually from the carrier, to unlock the card. This key is typically a six- or eight-digit code protected by a similar lockout policy, this time with a

ten-attempt limit. If this limit is reached, the card becomes permanently locked and must be discarded.

2.2.2 Forensic Interest.

SIM cards contain a wealth of information useful to forensic investigators [4]. While some of this data is accessible without authentication, much of it is in protected files and thus cannot be retreived without first entering a valid PIN. If a SIM is recovered while attached to a phone or mobile device, the PIN may be saved in the device and thus be retrieved by other means. However, in the case of SIMs recovered independent of a device, the PIN must first be obtained before accessing much of this data.

The National Institute of Standards and Technology (NIST) Systems and Network Security group divides available SIM forensic data into four categories ([27], [28]). A description of data available under each category follows.

2.2.2.1 Service-related information.

The first category is *service-related information*. This information is used to determine which network (or networks) the card connects to. The two identifiers of interest are the integrated circuit card identifier (ICCID) and the international mobile subscriber identity (IMSI). These numbers identify the SIM and the subscriber to the cellular service provider for authentication and billing purposes. The ICCID represents the card's unique serial number and cannot be changed. The ICCID is accessible without a PIN and is also typically imprinted on the card itself. The IMSI uniquely identifies the subscriber and includes fields for a country code, network code, and account number. Unlike the ICCID, the IMSI requires PIN entry for access. Additionally, the IMSI may be changed by the provider.

2.2.2.2 Phonebook and call information.

The second category, *phonebook and call information*, focuses on telephone numbers stored in the SIM. The two primary files used for phone number storage are the Abbreviated Dialing Numbers (ADN) and Last Numbers Dialed (LND) files. The ADN file contains

standard phonebook-style name-number pairs for easy retrieval and dialing. The LND file is used to record recent numbers dialed by the mobile device (regardless of whether the call was successful). Phones and devices are not required to use these files, however; many phones store phonebook and recent call information in internal device memory instead of on the SIM [51].

2.2.2.3 Messaging information.

The third type of forensic data contained within a SIM is *messaging information*. Short Message Service (SMS) messages are often saved to the SIM for later reference or retrieval. Both sent and received messages may be catalogued in the SMS file. Deleted messages may also be retrieved if they are only marked as deleted rather than being overwritten [28]. In addition to message text, message metadata is also stored, including timestamp and source/destination number. As with phonebook information, some or all of the SMS data may be stored in the phone's internal memory rather than the SIM. However, if present, it remains useful forensic evidence.

2.2.2.4 Location information.

Finally, SIMs contain *location information* within the Location Information (LOCI) file. At a high level, this file records the Location Area Code (LAC) for the group of network towers (called cells) with which the phone last communicated. These cell groupings are geographically based; thus, the LAC can be used to roughly identify the location where the device was last powered on and connected to a network.

2.2.3 Platform Implementation.

2.2.3.1 Platform overview.

Modern SIMs are complete embedded computer systems on a single chip. The typical SIM system includes a central processing unit (CPU), a cryptography processor, random-access memory (RAM), read-only memory (ROM), a strong random number generator (RNG), a communications interface, a clock generator, a power supply, and

busses connecting these components together. Note that ROM is often not truly read-only; in SIMs, it often takes the form of flash or electrically erasable programmable read-only memory (EEPROM), both of which can be erased and re-programmed. A block diagram of an example smart card processor (the Infineon SLE66) is shown in Figure 2.1.

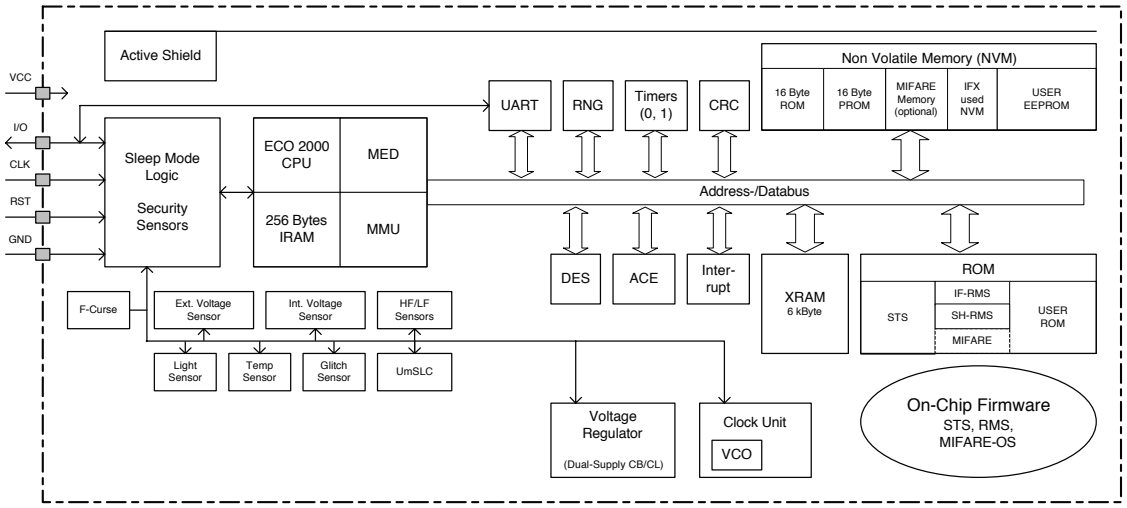

Figure 2.1: SLE66 security controller block diagram [23]

The complete system can be referred to as a system-on-chip (SoC) or microcontroller. While individual components such as the CPU, cryptography processor, or communications interface are discussed here as separate modules, it is important to note that they are all contained within a single semiconductor device.

2.2.3.2 Processor.

Early SIMs as well as modern low-end devices feature an 8-bit CPU. The Infineon SLE44 and SLE66 smart card controllers, for example, use an optimized derivative of the 8051 core with extensions to run some 16-bit instructions. [37]. Mid-range platforms such as the SLE76 use a true 16-bit platform, and the high-end SLE88 uses a 32-bit processor [25]. Other manufacturers such as NXP, Atmel, Samsung, and STMicrosystems have similar product lineups.

The CPU operates at clock rates from 3.57 MHz up to 66 MHz, depending on the model. A reference clock signal is generated by the host (i.e., the phone) and supplied to the SIM via an external clock input pin. The SIM receives this signal and uses a phase-locked loop (PLL) to multiply the clock to the required frequencies for the CPU, communications interface, and real-time clock (if present) [46]. The SIM may elect to stop or vary the frequency of these internal clocks in order to save power during inactive periods.

In addition to the main CPU, cryptography processors are used to provide hardware implementations of necessary cryptographic algorithms needed for the SIM's security functions. Most devices include the Data Encryption Standard (DES) and Advanced Encryption Standard (AES) algorithms, and high-end SIMs include Rivest-Shamir-Adleman (RSA) implementations as well. By offloading complex encryption work to this cryptography processor, the CPU is free to use its limited processing power for other tasks. Also, from a hardware security standpoint, it is easier to protect specific operations of a dedicated cryptographic device than it is to protect an entire general-purpose CPU.

2.2.3.3 Operating system.

The SIM operating system (OS) is different for every manufacturer, but most SIMs adhere to a specification called Java Card. The specification is maintained in the form of an international standard [11] as well as an application programming interface (API) by Oracle [42]. A Java Card-compatible OS is an application-based system that allows manufacturers, carriers, and other developers to quickly design applications that function on any SIM adhering to the standard. These applications are written in a subset of the Java programming language. As with Java on personal computers (PCs), a completed Java Card application is compiled into bytecode that runs on the Java virtual machine (JVM) provided by the host platform. Java Card-compatible bytecode is very similar to full Java bytecode, but is optimized for size, as Java Card platforms are often very limited in terms of available RAM and ROM.

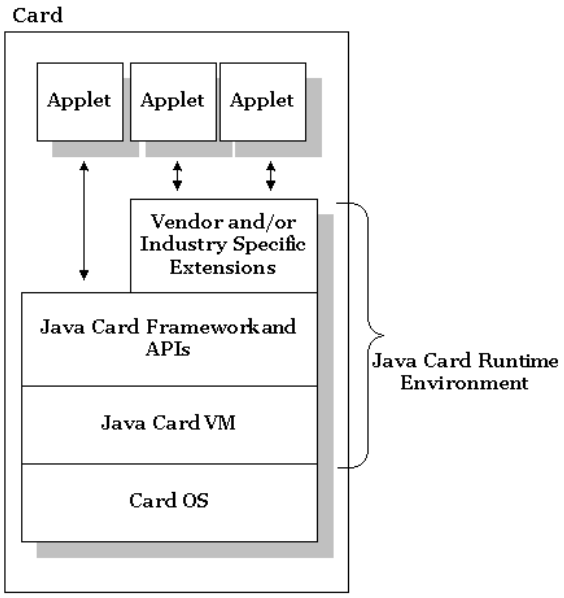

Figure 2.2: Java Card architecture and runtime environment [43]

The primary Java application running in the SIM's operating system is called USIM [50]. This application provides an interface for communication with the SIM and maintains the file system hierarchy. This hierarchy includes levels such as the master file (MF) and application dedicated file (ADF) as well as two file types, the elementary file (EF) and dedicated file (DF). These files contain a wealth of information related to the SIM, the phone, the network, and the user [5]. Data stored in these files includes the network encryption keys and certificates, user identity numbers, PIN codes, phonebook entries, text messages, and more; essentially all data retrievable from the SIM is accessed via one of these files. Each file has attributes that specify what privilege level is necessary to access the file [39].

2.2.4 Communication and Interfacing.

The SIM interface is a set of six contact pads on one side of the module, five of which are actually used. When the SIM is inserted into a device, spring contacts press against

these pads to form the electrical connections. This contact interface is standardized in [26] and is shown in Figure 2.3.

Figure 2.3: Smart card pin-out [7]

The power (VCC) and ground (GND) pads supply 3.3V power to the SIM. The clock (CLK) pad is an input for the SIM's reference clock, as the modules do not typically have an on-board oscillator. This is usually a 3.57 MHz input, however, some SIMs support faster clock rates. The reset (RST) pad is a standard CPU reset signal that controls the processor's operation. Finally, the I/O pad is the SIM's communication interface. This single pin provides an asynchronous serial port and is the only interface through which all communication to and from the card must take place. The programming (VPP) pad is used for manufacturer-specific flash programming.

When power is first applied to the card and the reset signal is de-asserted, the SIM sends an answer to reset (ATR). The ATR contains up to 33 bytes of card parameters, including maximum clock rate, communication baud rate, and protocol selection [26]. Apart from the ATR, all communication with the SIM takes place over one of two standardized protocols. These protocols are called T=0 and T=1.

In both protocols, communication between the USIM application and the host (phone) takes place in the form of messages called Application Protocol Data Units (APDUs). The T=0 protocol simply requires APDUs to be sent in a command-response sequence. While

the T=1 protocol also uses APDUs to hold data, it is a complex multi-layer networking-type protocol that will not be described here.

APDUs are either commands or responses. A command APDU contains the following fields:

- **CLA**: a one-byte field indicating whether the rest of the command follows the standard format or a proprietary format
- **INS**: a one-byte instruction opcode indicating which command should be executed
- **P1, P2**: two bytes containing arguments or parameters for the selected command
- **Lc**: 0-3 bytes defining the length in bytes of data to follow
- **Data**: Freeform data with length as described in Lc
- **Le**: 0-3 bytes defining the length in bytes of the expected response APDU

A response APDU contains only two fields:

- **Response**: freeform data with length as described by the received command APDU
- **SW1, SW2**: status of the requested command, i.e., success or failure (SW = "status word")

Examples of commonly used commands (specified by the INS field) include "select record," "get data," "put data," and "verify CHV". The last command is particularly of interest to this research: it is used to validate the PIN.

2.2.5 Security Features.

One of the primary purposes of the SIM is to provide security - Infineon even calls their smart card microcontrollers "security controllers." By necessity, then, SIM devices must have built-in security mechanisms to protect the secrets contained within the card. A brief overview of SIM-based authentication and encryption follows [3].

2.2.5.1 Protecting the keys.

In a mobile phone application using second-generation (2G) Global System for Mobile Communications (GSM) or third-generation (3G) Universal Mobile Telecommunications System (UMTS) networks, a challenge-response process is used to authenticate. The phone first sends identity information found on the SIM in the form of the ICCID and/or IMSI. The network provider's backend systems use this identity to locate the subscriber's record in a database and retrieve the subscriber's individual shared secret key.

Next, a random number is generated to form the challenge. This number is encrypted with the shared secret key to calculate the expected response. The challenge is then transmitted to the phone, which in turn presents it to the SIM. The SIM also contains the shared secret key (pre-loaded prior to sale by the provider) and uses it to calculate the response, which is transmitted back to the network provider for verification.

Once authenticated, the randomly-generated number as well as the shared secret are used to calculate a session key that is used in ongoing encryption of over-the-air communication for the rest of the session. While the session encryption itself is performed by the phone, the key calculation process is performed by the SIM.

Even with this simplification of the network security process, the need to protect the shared secret key is immediately apparent. This shared key (called the *Ki*) can be loaded into the card but not retrieved [13], and the SIM never transmits it to the host device. Exfiltration of the key would allow theft of service, identity spoofing, and eavesdropping. It could also allow multiple devices to use the same account.

Due to this dual concern of both the customer's privacy and security as well as the provider's billing integrity, it is in the best interest of the provider to select a high-security SIM card platform. As a result, even low-cost SIMs are designed with security in mind.

At a high level, the closed nature of the SIM system enhances security. While a personal computer allows custom application execution, network connectivity, and direct

access to storage, the SIM has a much simpler interface to its surroundings. All communication to the SIM must take place over its universal asynchronous receiver/transmitter (UART) interface via a defined set of commands. Arbitrary application code cannot be executed by the user, and a PIN-locked card blocks all but the most basic status commands.

2.2.5.2 Software-based protection.

The SIM file system, as described earlier, has security attributes to prevent unprivileged users or code from accessing critical files such as those containing the PINs or the Ki. Privilege levels protect these files on an operating-system level, and the virtual machine-based nature of the Java language prevents bypassing these restrictions with raw memory access.

Even the specifics of the algorithms used to perform GSM authentication and encryption were kept secret until they were reverse engineered by Marc Briceno in 1999 [6], revealing weaknesses in the algorithm's implementation that allowed for easier SIM identity cloning. Modern algorithms used for UMTS and Long-Term Evolution (LTE) networks are standardized and published [12], revealing implementation details for the presumed benefit of stronger cryptography.

Unfortunately, PIN authentication algorithms are not published, as from the standard's perspective, the SIM needs only to validate the PIN as supplied in plain text. Any encryption or obfuscation of the PIN in storage or validation is up to the SIM manufacturer or integrator. While weaknesses may exist in any particular implementation, the "black box" nature of this component makes vulnerability analysis difficult.

2.2.5.3 Hardware-based protection.

In addition to protecting against software- and cryptography-based attacks, the hardware itself is protected against access and manipulation. A full-size embedded or desktop computer has accessible busses on which data flowing between the CPU, RAM, and storage can be observed. Generally speaking, any data processed by the CPU must

pass over this bus and is thus vulnerable to observation. Andrew Huang famously used this technique to obtain the code signing key from the Microsoft Xbox gaming console in 2003 [22].

As embedded computing platforms, it follows that SIMs should be vulnerable in the same way, and strictly speaking, they are. However, unlike a discrete embedded platform where the 16 or 32 physical bus tracks on the circuit board may occupy 2 cm^2 or more, the die containing the *entire* SIM system (busses, processors, memory, etc.) may occupy 4 mm^2 or less. This tiny die is also surrounded by an encapsulating compound that must be removed with fuming nitric acid or a similar caustic chemical. Once the die is bare, the problem of extreme miniaturization can be mitigated with the use of a focused ion beam (FIB) workstation, but the process is costly and prone to error [46]. Christopher Tarnovsky of Flylogic Engineering notes in [52] that gaining access to the bus of the Infineon SLE66 controller was a nine-month process. Several chips and probes were destroyed in the process, and the use of the FIB equipment cost as much as $350 per hour.

However, despite the difficulty, any well-equipped FIB lab is eventually able to repeat this process reliably; if the chip is successfully decapsulated, the bus may be read while the device is running (if possible), or the storage memory/ROM can be powered up and read out independently. A simple protection mechanism to discourage a probing attack is to place a fine mesh on the topmost metal layer of the chip. This mesh may have one or more "channels" of fine traces blocking access to the circuitry below. If power is applied to the SIM after cutting or removing the mesh, the device's security mechanism can either put the SIM into a temporarily disabled state or cause the device to permanently erase its memory [16]. An example of a simple single-channel mesh design is shown in Figure 2.4.

While it is exceedingly difficult, tedious work with a FIB workstation can remove these meshes and attach fine wires to bypass their security mechanisms [52]. Therefore, the chip must therefore be further protected given the assumption that busses are *somewhat*

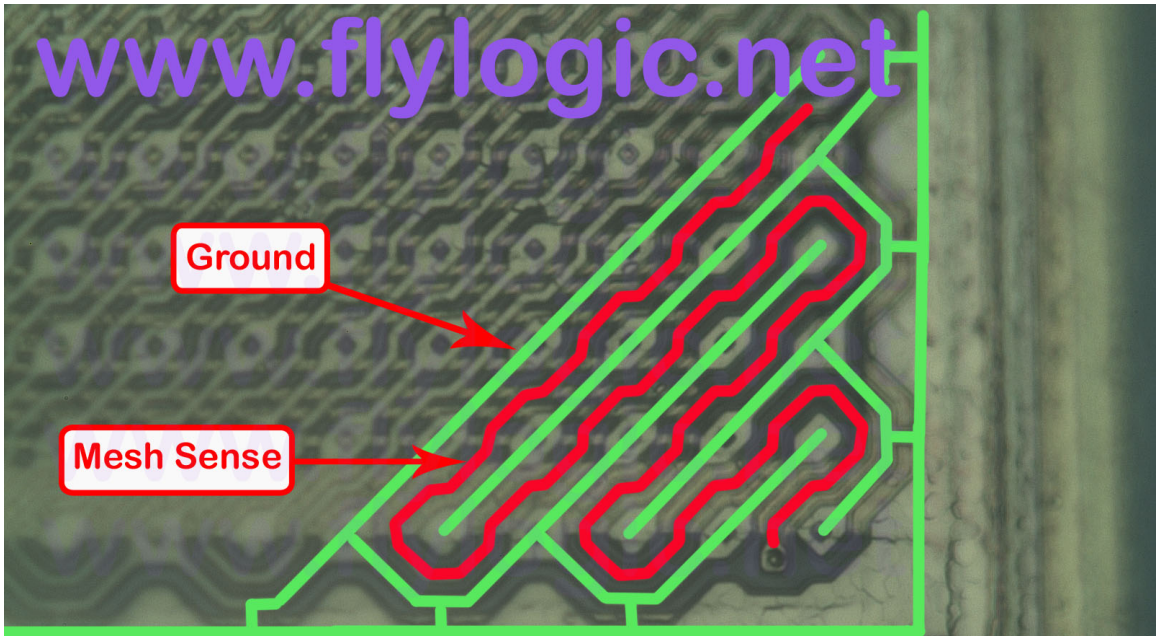

Figure 2.4: ST16601 simple single-channel mesh design [15]

freely accessible. One simple obfuscation is bus scrambling. Outgoing bus lines from the CPU are fed into a matrix, and all but one connection for each bus line are laser-cut in a random configuration at the time of manufacture. The result is that the bus lines for any given SIM are randomly scrambled, and two seemingly identical SIMs are unlikely to have identical bus configurations. While simply an obfuscation, this can slow down a probe-based attack and may have implications for the electromagnetic signature of the device as well [46].

Another bus protection mechanism is bus encryption. By placing a memory encryption device between the CPU and its peripherals, the code and data passed on the bus and stored in RAM and ROM are encrypted [24]. This prevents a bus-snooping attacker from determining the operations of the SIM even if the aforementioned scrambling is defeated. Additionally, this encryption can be performed with a unique key for each SIM device, causing identical bus traffic to appear dissimilar between individual SIMs.

Other physical attacks include optical fault injection attacks and glitching. Optical fault injection involves decapsulating the chip and using a laser to switch transistors on by photon bombardment. A well-placed pulse can actually cause bits to flip in such a way that security measures can be bypassed. Glitching involves continuously or momentarily operating the SIM at voltages or frequencies just outside its specifications. If unexpected behavior is introduced, it may be possible to cause the SIM to enter an unlocked state. SIMs are protected against such attacks through on-die light sensors, voltage sensors, frequency monitors, and glitch detectors.

Finally, the SIM is protected against more than direct, invasive physical attacks. Side-channel analysis is a non-invasive, observational attack vector that does not require decapsulation and FIB probing. Instead, the SIM is operated in standard ways and the timing, power consumption, and electromagnetic emanations are observed. Careful measurement of these parameters can reveal details about the operation of the SIM. An in-depth look at side-channel attacks and countermeasures follows in the next section, but at a high level, the SIM must include mechanisms that hinder these attacks.

In summary, the SIM is built to be a secure device from the silicon to the software. Most of the security mechanisms discussed are intended to protect cryptography-related data and operations, as these components ensure confidentiality and integrity of the user's account and network data. However, the side effect – certainly not an accidental one – is that the SIM's security-minded design makes it very difficult to extract the PIN and obtain access to the user's stored data.

2.3 Side-Channel Analysis (SCA)

2.3.1 SCA Overview.

The term *side-channel analysis* refers to the analysis of data transmitted or "leaked" from a device via an unintentional channel. A SIM card, for example, only has one intentional channel for sending data to its host: the serial port. Higher-end smart cards

may have an additional Universal Serial Bus (USB) or contactless channel as well. These interfaces form primary channels, and all data leaving over these channels – whether authorized or not – is legitimate from a side channel analysis perspective. Information that leaves the device through anything other than these intentionally defined channels is said to *leak* through a *side channel*. It is important to note that side channel leakages can include both data that is normally transmitted over a legitimate channel (such as the letters pressed on a computer keyboard) as well as data that is never transmitted by the device (such as the GSM Ki). Also, information leaving the device is not purely data; while data is often the target (such as an encryption key), side channel leakage of a device's instruction state is also a vulnerability. One such vulnerability occurs if a side channel leaks the fact that the device has rejected a password and is about to decrement a retry counter. This information could allow the hacker to cut power to the device before the counter is written.

Common side channels include operation timing, power consumption, and electromagnetic emanations. A brief description of each follows.

2.3.2 Types of SCA.

2.3.2.1 Timing analysis.

One of the earliest and most easily understood side channel attacks is the timing attack. A timing attack occurs when side channel leakage is obtained by observing the amount of time an operation takes to occur. An example of this is found in a simple string compare such as might be used by a password-checking function. The system loads both strings into memory and sequentially compares each byte. As soon as a comparison fails, the loop exits and the function returns the result that the strings do not match (or the password is invalid). However, if every compare succeeds and the end of the string is reached successfully, the function returns the result that the strings match (or access is granted). The number of comparisons the function performs before it returns its result is directly related to the number of matching bytes in the strings. Now, suppose that each byte comparison

instruction takes 1 microsecond (μs) to complete. Comparing strings in which the first six bytes match takes 4 μs longer to return than if only the first two bytes match. A password validator that operates in this way (using a standard string-compare) is guaranteed to be vulnerable to this simple timing attack, as a brute-forcing attacker is given information about his progress that drastically reduces the password search space.

Paul Kocher in [35] is generally cited as the earliest published demonstrator of a timing attack on cryptosystems. Kocher bases his attack on the fact that computationally expensive cryptographic functions often run in variable amounts of time. In the simplest case, he demonstrates the attack on the binary modular exponentiation method. This is a repetitive algorithm that, in each iteration through its loop, either performs or skips a modulus operation based on the value of each bit in the input. The bit value that causes the modulus to execute takes more time than the value that causes the skip. Thus, by trial and error, an attacker can eventually figure out the entire value. Kocher goes on to explain how this principle can be applied to various cryptosystems such as RSA and Diffie-Hellman.

Kocher's ideas are used by Dhem et al. [10] to create a working attack against RSA in early smart cards, resulting in a modification of the implementation in that particular smart card standard. As a "practical" attack, this paper shows that timing attacks are a true concern in cryptosystems and not simply a theoretical danger. It is noted that the attack fails against RSA implementations using the Chinese Remainder Theorem, but Schindler in [48] provides a timing attack that solves this problem and is even more efficient than the original attack.

2.3.2.2 Power analysis.

Power analysis attacks are another means of side-channel analysis. Power analysis relies on the fact that current flows across transistor junctions when a given transistor enters a switched-on state. Thus, the device's operation is observable through changes in the overall current draw: each transistor in the on-state increases the required current,

and transistors in the off-state have the opposite effect. Small current consumption changes like these create detailed pictures of low-level data, instruction, and execution processes occurring inside the device. Additionally, as different modules of a complex microcontroller switch on and off, observed patterns and changes in a higher-level "zoomed-out" view of the power consumption trace shows the large-scale operations over time [31].

Power analysis requires that the current draw of a device be measured. This is achieved by placing a small-valued resistor in line with the device's power input (either on the voltage or ground line). The voltage drop across the resistor is then measured via a traditional oscilloscope probe and Ohm's law is used to convert this voltage to current ($I = \Delta V/R$). As the process requires the cutting of a trace or wire, this "power tap" is considered invasive but does not require extensive modification.

As with timing analysis attacks, Kocher et al. provide the earliest studies on power analysis attacks ([33], [34]). He divides power analysis attacks into two types: Simple Power Analysis (SPA) and Differential Power Analysis (DPA). Simple power analysis refers to cases where the power consumption trace exhibits a high-level instruction dependency, that is, the data fed into an algorithm determines the execution path. The application of SPA is very similar to that of timing attacks in that decisions in the algorithm's flow (such as a conditional branch) cause information about the protected data or key to leak. The paper shows how the sixteen rounds of a DES operation is easily identified by its repetitive pattern of power consumption. It also notes a few conditional branches that make DES susceptible to SPA.

Differential power analysis is used when simple power analysis is insufficient to extract the required data from the device. Unlike SPA, where a single trace or a small number of averaged traces demonstrate a clear representation of the data or operations, DPA requires larger numbers of traces (1,000, 10,000, 100,000, or more). Statistical analyses are

then conducted on the data in conjunction with generated intermediate data for many key guesses. Averaging is performed across the traces and a classification is given to each trace for each key guess. The correlation value is calculated for each guess and the guessed key with the highest correlation – the DPA peak – is chosen to be correct. This type of attack does not require knowledge of the implementation but does require some knowledge of the algorithm in place [1].

Kocher's introductory DPA work is largely theoretical, but more practical attacks appear in [32]. Specifically, attacks against DES and RC4 are shown. In addition, the use of Hamming weights as a side-channel goal is introduced. The Hamming weight of a number is simply the sum of the set bits: for example, the Hamming weight of 4 (0100) is 1, and the Hamming weight of 7 (0111) is 3. The number of set bits directly correlates to power consumption and provides a useful tool in DPA. While an attack providing the Hamming weight of all or part of the key does not completely "crack" the given encryption, it can substantially reduce the keyspace and allows for smaller and more manageable collections. Hamming weights are also usable as a DPA subkey power model [31].

In addition to data leakage, power analysis also provides a means of observing hardware states in order to exploit a device. For example, a SIM card must maintain a retry counter to allow only three invalid PIN attempts before locking to the PUK. A naïve implementation of this function (used in early smart cards [46]) decrements the counter each time an attempt fails, locking the card when the counter reaches zero. However, the hardware implementation provides advance warning that this decrement is about to occur. The PIN retry counter must be stored in non-volatile memory in order to persist. The types of non-volatile memory used in SIMs include EEPROM and, more recently, flash memory, both of which draw an increased amount of power when entering a write cycle. By watching for the leading edge of this power spike and immediately cutting power to the SIM, the counter is never updated and the PIN is breakable by a simple brute-force attack.

2.3.2.3 Electromagnetic analysis.

Electromagnetic analysis provides yet another means of side-channel attack. Instead of capturing the power consumption of a given device, the electromagnetic (EM) emanations are measured. Complementary metal oxide semiconductor (CMOS) devices, which include most semiconductor devices today, consume power during gate transitions from high to low levels. Additionally, the act of switching a gate causes a tiny short circuit and a small spike in current draw as the signal propagates through the gate's transistors. All of these changes in flow of power alter the electromagnetic field surrounding the device, and these changes can be picked up by a special probe.

Capturing EM data requires a near-field probe which contains a simple coil of wire. This probe is placed very close to the device being analyzed. Current moving within the device induces small currents in the coil, which are then amplified by an attached low-noise amplifier (LNA) and captured by an oscilloscope. This captured data is analyzed in much the same manner as in power analysis attacks, both because electromagnetic emanation is closely related to power consumption, as well as because the statistical methods used are effective for both types of analog signals [38].

Electromagnetic analysis has some advantages over power analysis, however. Unlike DPA, it is a non-invasive attack, eliminating the need to modify the target. Second, the nature of using a movable probe results in the ability to perform localized captures. In other words, if the CPU component of the SoC is found to leak the most information, the probe is positioned directly over that area of the die to obtain the most signal and the least noise. Power analysis, on the other hand, is limited to the aggregate power consumption of the entire device. Finally, electromagnetic analysis (EMA) allows the observation of the charge and discharge cycles of a gate discretely, providing more detail on each gate's transitions than power analysis [44]. This allows, for example, a bit's transition from 1 to 0

to be more readily distinguished from a 0 to 1 transition, and thus, is a stronger side-channel leakage.

While Kocher's DPA papers mention electromagnetic analysis from a theoretical standpoint, it and other papers shortly following it lack actual experimental results. Gandolfi et al. in [19] demonstrate some of the first true results of successfully attacking DES, COMP128, and RSA with EMA. The paper notes the importance of selecting a physical location on the chip where activity is highest during the desired process, a methodology also followed by this thesis research. It also provides a detailed comparison of DPA vs. Differential Electromagnetic Analysis (DEMA), the latter term coined along with Simple Electromagnetic Analysis (SEMA) by Quisquater et al. to match Kocher's DPA and SPA counterparts, respectively [45]. It is worth noting that even though EMA is found in some cases to be superior to power analysis, DPA statistical techniques are applicable to EMA attacks [34].

Mulder el al. in [8] show that hardware cryptographic modules – such as those found in smart cards – introduce another layer of complexity as encryption operations occur in parallel instead of sequentially. This greatly increases the number of transitions happening simultaneously, thus muddying the signal and requiring much higher amounts of data to break. The example 160-bit hardware cryptosystem requires 20 times more collections to obtain the key than the same system in software on an 8-bit smart card.

Other challenges of EMA include finding an appropriate probe, filtering out information outside of the desired clock frequency, and locating the area of maximum leakage. With a high resolution probe, very minor changes in location can select completely different emanations [9]. Electromagnetic analysis proves effective, but assembling an attack on a new system is quite a challenge.

2.4 Side-Channel Analysis Attacks on Subscriber Identity Modules

2.4.1 Attacks on encryption.

The majority of attacks on SIMs focus on acquiring the Ki, and thus attack the encryption functions of the card rather than the PIN. While acquiring the Ki is not the goal of this research, methods used in such attacks provide useful information on side-channel analysis of SIMs and countermeasures present therein.

Prior to attacks on SIMs specifically, Side-Channel Analysis (SCA) attacks against smart cards exist as early as the first year of DPA's public existence. Since the smart card is a closed-system cryptography system with widespread use, it is logically one of the devices most affected by SCA and thus the target of much research. In one of the earliest papers on the topic, Messerges et al. in [40] demonstrates the vulnerability of simple implementations of DES to SPA. It also shows the vulnerability of better implementations to DPA given approximately 1,300 traces with randomly-selected inputs.

DPA is used in [47] to adaptively attack the COMP128 algorithm in a SIM. Initially, a standard DPA attack is used with 1,000 traces, and DPA is performed against all 256 key byte hypotheses. This attack fails and further analysis is necessary. Several assumptions about implementation details provide a reasonable hypothesis for the inner workings of the implementation, and further classification is used to group the traces based on a variety of factors until the assumptions are shown to be correct. Random inputs to the device (that is, having the SIM encrypt random values) can find the key with these methods in only 500 traces. However, an adaptive search increases the attack speed considerably. By taking a single collection with a single input value and then performing DPA on that data, each subsequent input value is chosen to quickly restrict the possible outcomes one bit at a time. Continuing with this approach, the attack consistently succeeds after only eight collections. The number eight is interesting here as most DPA approaches require well upwards of 100 traces, while eight traces is less than the ten-attempt PUK lockout limit.

Another COMP128 attack is performed by Novak in [41] based on a vulnerability in the algorithm's table lookup. This vulnerability splits the key space in half as the implementation on an 8-bit processor requires the lookup table be split for indexing. Yet another DPA-based attack is presented by Zhou et al. in [55]. In both of these cases, and particularly in the latter, it is observed that the attacks depend on the 8-bit nature of the controller. However, the carrier phase-out of 8-bit SIMs began in 2003 [55] and has likely been long completed at the time of this writing ten years later. The use of a 16-bit CPU entirely removes the need to split the lookup table and, as such, eliminates the possibility of an attack through these routes.

2.4.2 Attacks on Personal Identification Numbers.

Note in each of these above cases, a relatively large number of encryptions are performed and some knowledge of the encryption algorithm used is required. Unfortunately, the high-level implementation of the PIN verification procedure confounds both of these expectations. First, the SIM lock-out counter only allows the acquisition of two traces (in the case of the PIN) or nine traces (in the case of the PUK), which does not allow correlation or clustering to be performed. Second, the algorithmic uncertainty makes it very difficult to determine intermediate values that may be passing through different parts of the circuit. The PIN may or may not be encrypted, may or may not be obfuscated, and may or may not be checked multiple times.

There are very few research papers specifically studying the PIN of SIM cards. Folkman in [18] identifies DES encryption being used as part of the PIN verification process on a low-end 8-bit SIM. The SIM appears to have little or no countermeasure protection aside from the use of encryption, as many identifiable patterns and processes are clearly visible in captured power traces. The goal of encryption in PIN verification is to obfuscate exactly where a comparison is failing. Instead of comparing a four-digit PIN byte-by-byte in the clear, the stored encrypted PIN is compared to an encrypted version of the input PIN.

Both encryptions may be salted with additional data such as the card's unique identifier. When the encrypted values are compared, the position at which they fail should reveal no information about where the stored and input PINs differed [14].

In this case, however, it appears that the encryption actually introduced a new vulnerability while attempting to obscure another. The captured power traces clearly show the rounds of the DES operations, and once these are identified, standard DPA or even SPA attacks can potentially be used against this data. Unfortunately, the papers cited do not provide results as to whether or not the attacks were successful; the papers focus on the reverse engineering of the process more than the recovery of the PIN.

2.5 Countermeasures

2.5.1 Inadvertent countermeasures.

The complex SIM platform provides a number of features that inadvertently hinder or foil side-channel attacks. Countermeasures discussed in this section are not strictly countermeasures; rather, they are platform or implementation details that happen to make the SCA process more difficult.

The first of these is simply identification of the specific make and model of the SIM at hand. There are many different cards available with many combinations of security features, cryptography implementations, and even software configurations. While manufacturers frequently advertise the available features in product briefs and short datasheets, it is very difficult to identify the features, chipset, or even manufacturer of a given SIM. SIMs are typically re-marked with a carrier's custom product number and cross-references are not available.

Related to this is the lack of understanding of the PIN validation algorithm. This could be considered both an inadvertent and an intentional countermeasure, as the details of the algorithm are probably protected for security, but would likely have no reason to be published even if it was not a security risk.

One way to identify a SIM is by the pattern formed by the contacts on the bottom of the card [46]. This pattern is typically unique to the manufacturer and occasionally to individual models of SIM. Rankl identifies over 400 unique patterns but does not name the manufacturers associated with each. Still, if evidence is provided that a given card is, for example, a Gemalto SIM, other Gemalto SIMs can likely be identified by the same pattern. Such is not usually the case for identifying models or security features, however.

Another inadvertent countermeasure is the use of the Java Card operating system. In recent versions, the operating system supports multithreading and features a task scheduler that could potentially interrupt a cryptography operation or SIM verification process in varying locations such that multiple traces of the same operation appear different. Not only do the execution interruptions shift the SIM-verification operations around temporally (which may be bypassed by elastic alignment techniques [54]), but they also introduce other miscellaneous data that can confuse the correlation process.

2.5.2 Intentional countermeasures.

Smart card manufacturers have been responding to side-channel analysis attacks since Kocher first published in the late 1990s. ST Microelectronics reassures its customers in a 1998 white paper ([49]) that their smart cards are strongly protected against SPA and DPA in a number of ways. One of these ways is a simple interrupt system with what ST calls an unpredictable number generator. The idea is that interrupts set to occur at random intervals cause unpredictable timing and power consumption fluctuations to discourage attacks. The paper also mentions a software-adjustable system clock, which can presumably be varied during critical operations in such a way as to randomize execution times without interrupts. Finally, the paper includes one of the earlier mentions of "tripwire" capability: if certain invalid conditions (such as unexpected states, bad opcodes, etc.) are met, the device is capable of erasing its entire RAM and ROM. While not comprehensive, these early attempts at avoiding DPA establish a baseline for modern SIM card security features.

The clock is a key component in countermeasures as it is the heartbeat of the device. Random clocking as mentioned above is substantially more effective than random software-based delays, and by clocking the serial port (which requires a fixed clock) independently, it is not a costly design change ([36]). In addition to the smart card varying the clock on its own, sensors exist to monitor clock fluctuations outside of this range. If the clock rate drops excessively low – a technique usable by an attacker to more easily observe transitions – the device is reset.

Shielding, as discussed in an earlier section, can prevent probing in the case of device decapsulation [36]. However, a shield or mesh also acts as an electromagnetic shield for signals emanating from the device. In some cases, lines of this shield are connected to a random noise generator [53]. This noise generator, in addition to contributing to random power consumption, generates a constantly varying signal that is received by the side-channel probe. Unwanted variations such as these confuse the side-channel attack and prevent reliable correlation. As an active shield, the SIM monitors the signal flowing through the shield to ensure its presence and shuts down (or even erases) if the shield is removed. Light sensors act as similar tripwires to prevent exposed dies from operating and/or laser-based fault injection attacks [46].

Busses in smart cards can easily be scrambled at the time of manufacturing [46]. A matrix hidden somewhere in the die is used to join address and/or data lines in non-linear order such that the resulting bus is not plainly understandable. While this feature is intended as a probing countermeasure, if each SIM uses a different scrambling matrix, this slows down the potential attacker from readily interpreting instructions or data between SIMs. An example of this bus scrambling technique is shown in Figure 2.5.

Finally, memory encryption is a common hardware SCA countermeasure. By standing between the system's memory and the processor, a memory encryption device prevents simple interpretation of memory bus leakage [24]. Encryption of these busses and

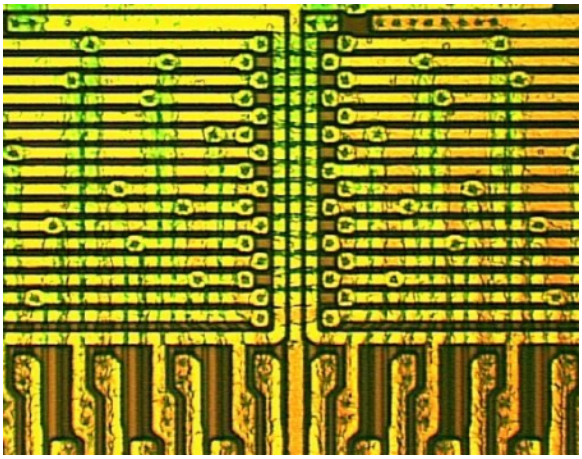

Figure 2.5: ST16F48A bus scrambling matrix [36]

cryptography-related registers corrupts expected Hamming weights from appearing and lessens the information leakage even if electromagnetic emanations are reliably detected [2].

Not all countermeasures require complex hardware redesigns however. Hundreds of implementations of DES, COMP128, RSA, and AES exist with various side-channel countermeasures. Execution times are kept constant by providing identical do-nothing branches to complement branches that have purposeful operations. Table lookups are scrambled or performed in obfuscated ways. Unneeded portions of random or scrambled data are processed to distract analysts.

While not documented explicitly in datasheets, a number of countermeasures are mentioned in patents by their various manufacturers. A manufacturer's ownership of patents does not ascertain their use of a certain countermeasure technology, but a datasheet bullet point may be correlated with a patent for implementation hints. For example, some patents mention the PIN specifically, discussing the verification of a fake PIN [29] or storing a second PIN to encrypt and compare against [30].

2.6 Summary

SIM cards are a secure platform for protecting sensitive information. Forensic data contained therein cannot be easily extracted via intended channels without authorization. Side-channel analysis is a powerful means of extracting information not meant for the outside world, and is effective at obtaining encryption keys from a variety of systems. SIMs have a number of countermeasures in place to prevent leakage of protected information through these side channels, but few documented attempts at attacking the PIN exist, leaving side-channel analysis as a potential means of PIN extraction.

III. Methodology

3.1 Introduction

This chapter presents the methodology used to evaluate the performance of side-channel analysis attacks applied to SIM cards. First, the problem definition, goals and hypothesis, and high-level experimental approach are discussed in Section 3.2. The side-channel analysis process is defined and bounded as a system in Section 3.3. Next, the problem is defined in terms of services provided for a workload in Sections 3.4 and 3.5. Expected outcomes are described as metrics in Section 3.6. Factors and parameters are discussed in Sections 3.8 and 3.7, respectively. A complete description of the equipment used, experimental setup, and evaluation process is presented in Section 3.9. Finally, the experimental design is covered in Section 3.10.

3.2 Problem Definition

3.2.1 Goals and Hypothesis.

The ultimate goal of this research is to develop a functional electromagnetic side-channel attack against a SIM card. Verified, consistent correlation of captured EM data with stored PIN values enables the construction of a side-channel analysis system to be used against arbitrary SIM cards (within the subset of makes and models tested). A successful system can repeatably recover an unknown SIM card's PIN.

Additionally, this research evaluates the effectiveness and impact of manufacturer countermeasures against SCA attacks. Since side-channel attacks are well-known in the field, specific protections are in place on most cards to protect against information leakage. These protections or countermeasures are typically designed to obscure the card's cryptographic operations as opposed to the PIN. The efficacy of such methods in protecting the PIN is shown by this work.

Finally, this research tests existing collection techniques, SCA attacks, and correlation techniques against a SIM card PIN application. The reproducible result of unlocking a SIM with an unknown PIN validates this use of the selected methods of SCA.

The working hypothesis for this research is that electromagnetic side-channel analysis methods can successfully identify the PIN (or PUK) of a locked SIM.

3.2.2 Approach.

At a high level, the approach is divided into two phases, each of which is repeated with various combinations of factors as explained in subsequent sections. The first phase is known as the "training" phase. A given SIM is placed in the test apparatus for evaluation under a given set of parameters. The data from this collection is evaluated to observe necessary correlations. Given that the card is deemed "leaky" enough and the collection method is found to produce useful data, a second and more in-depth collection is taken. It is expected that, in a production system, this training process is repeated for each new make and model (and likely revision) of SIM presented to the system. This training enables the system to recognize the card's discrete states and the data passing through its internal components.

Given that the first phase is successful, that is, usable training data is collected, a second phase is executed. A representative unknown PIN card is presented to the system and evaluated using the same apparatus as in the first phase. This simulates an operational perspective: the use of SCA techniques to break an unknown card. A comparatively small collection is taken from the card such that its retry limits are not exceeded (either two or nine tries to attack the PIN or PUK, respectively). Finally, a correlation algorithm classifies the dataset and determines the most likely stored PIN.

Attacks carried out are all electromagnetically based, but the correlation work is performed by power analysis algorithms as shown in [39].

3.3 System Boundaries

The system under test (SUT) is the SIM Side-Channel Analysis System, a combination of commercial off-the-shelf (COTS) and custom-developed components assembled to automate much of the side-channel attack process. At the highest level, the SUT encompasses the SIM card itself, the process automation and data collection system, and the correlation and classification system as shown in Figure 3.1. The focus of the work is on the development and manipulation of the latter two components, although factors affecting all three are varied throughout the process. While the SIM itself is part of the system under test, variations in card make, model, and stored PIN code are all considered part of the workload.

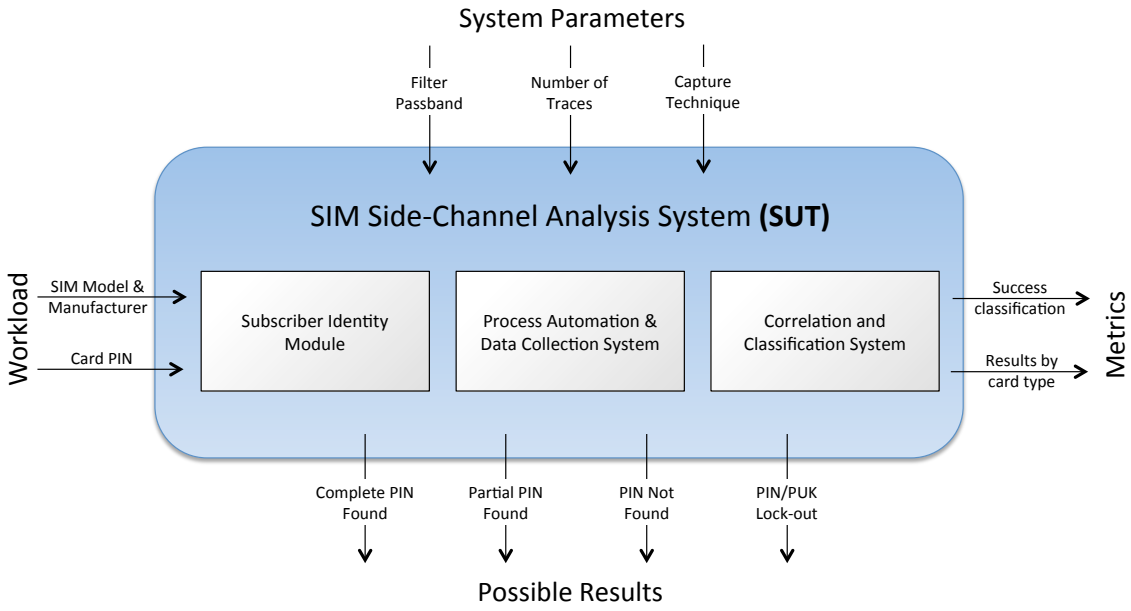

Figure 3.1: SIM Side-Channel Analysis System (SUT)

This research does not consider other types of smart cards or identity modules, although it could be adapted to do so. Many other smart card-based authentication systems use a PIN code and would be ideal candidates for further research. Additionally, the system

only considers leakages acquired and analyzed through electromagnetic analysis; other forms of side-channel leakage, while potentially interesting, are not considered.

3.4 System Services

The primary service provided by the SIM Side-Channel Analysis System is the recovery of unknown PINs. The service requires captured data from the Data Collection component whitch implicitly includes the desired unknown PIN (whether or not the PIN is recoverable). The processing of this data is completed by the Correlation and Classification component, which is used to provide the final results. Provided that sufficient correlation is found to make correlation and classification decisions, the output of the service is one of the following:

- Complete PIN found – the best case scenario; the PIN has been found and all digits are correct.
- Partial PIN found – not enough correlation to identify all digits, but others are known.
- PIN not found – not enough correlation for any digit to be determined consistently.
- PIN/PUK lockout – the card is locked because the retry count has been reached.

The Data Collection component provides captured data to the Correlation and Classification component. The outcome is simply the requested data, if it exists.

The SIM provides a number of services to a cellular device as discussed in Chapter 2. In the bounds of this research, however, it simply provides a communication interface to validate PIN codes. Its outcomes consist of the following:

- PIN good – the provided PIN matches the stored PIN
- PIN bad – the provided PIN does not match the stored PIN
- PIN lockout – the retry limit was reached and further PINs will not be validated

3.5 Workload

The workload of the SIM Side-Channel Analysis System consist of SIMs along with their unknown PINs. Each workload "unit" – that is, a single SIM with an unknown PIN – is processed individually, so there is no concept of workload throughput.

There are two workload parameters, both of which affect the performance of the system. The first of these is the make and model of SIM. Card features, implementation details, and manufacturing differences are all likely to affect the SCA process and therefore must be considered as a parameter. Note that in this research, any given unique make and model pair is considered independent from any other make-model pair. As multiple SIMs may share chipsets, cell-level components, or electromagnetic signatures, the examination of similarities and differences between SIMs of same and different manufacturers is a topic for future research.

The second workload parameter is the value of the stored PIN itself. Due to differences in current consumption between a bit being set or cleared – some of the principles the EMA process relies on – the captured trace data will differ based on the value of each PIN digit. Depending on countermeasures and architecture design, certain PIN codes are more likely to be detectable than others, thus, this parameter is likely to affect workload results.

3.6 Performance Metrics

Performance of the system is measured primarily against the desired output classification. The possible results are fourfold, and as such, percentages for the following four result categories are desired:

- Complete PIN found
- Partial PIN found
- PIN not found
- PIN/PUK lockout

Judging between these results, it is easy to ascertain the effectiveness of the technique as the metrics correlate directly. It is intuitively apparent that the desired outcome is for the "complete PIN found" percentage to be the highest. However, the remaining percentages remain useful, as they help identify partial successes and may guide assisted brute forcing.

The success metric is also ranked in terms of workload parameters. Since these workload parameters affect the system's operation, they have an effect on the output and must be observed. This ranking helps to determine, for example, collection techniques that are more effective at higher trace counts versus those effective at lower counts.

3.7 System Parameters

Since this research deals with a SIM card outside of its normal operational environment – that is, the SIM is attached to the experimental apparatus apart from a cellular phone), many of the normal operational parameters (such as battery power, radio frequency environment, and frequent communication with the phone) are removed from study. However, the overall SUT has several parameters of note.

The first of these is the filter frequency. Since EMA captures electromagnetic frequencies, noise is a large problem that must be dealt with. The first line of defense apart from the physical probe shield is in-line filtering. A well-selected filter will exclude frequencies outside of those produced by the card in normal operation and only allow desirable data to pass into the collection system. The selection of this range, or passband, is highly important, as an incorrectly centered or overly narrow passband will filter out useful leakage information.

Another parameter is the number of traces captured. Many traces must be captured to average out noise, transients, and the like. A large number of traces results in impractical amounts of data for analysis; additionally, a real-world SIM application is useless unless the number of required collections does not exceed the retry limit. However, too few trace collections will not produce a useful level of correlation.

Finally, the selection of a capture technique is important. The mechanical layout and structure of a given SIM as well as the use of shielding determines the location and direction of information leakage. By capturing from the top and bottom of the SIM as well as from both directions together in a novel noise cancellation technique, the chance is higher that one of these methods finds usable leakage. Capturing in additional planes (such as from the sides) as well as at varying angles is a viable technique for future work, although it requires the use of a more dextrous positioning assembly such as a robotic arm.

3.8 Factors

The following factors are selected out of the given workload and system parameters to provide the best results given the least number of experiments. Parameters not listed here, such as filter frequency and card make and model, are set at a constant level throughout all experiments. Factors are summarized in Table 3.1 below.

Table 3.1: Factors and Levels

	Level 1	Level 2	Level 3
Collection Technique	Top	Bottom	Dual
Number of traces	10	1,000	25,000

The number of traces collected is a very influential factor in the experiments as discussed earlier. Levels of 10, 1,000, and 25,000 are chosen to more fully evaluate other factors. Collecting 25,000 traces initially provides a platform in which even small levels of correlation can likely be determined. This is necessary for the training stage and a correlation and classification system that captures no correlation at this high number of traces can be discarded. A level of 10 traces tests performance near real-world levels, as useful extraction of a PIN or PUK requires a number of traces less than the lock-out counter

(two traces for the PIN, nine traces for the PUK). A collection of 1,000 traces is chosen as an intermediate value between those discussed previously.

In addition to the number of traces, the collection technique is varied among three levels. Collection from the top of the card is the traditional collection method and is also the simplest. Collection from the bottom possibly provides some benefit if the top of the die is covered by a shield mesh. However, the bottom of the die is covered by the SIM's interface contacts; specifically, the center ground terminal blocks the entirety of the die area. Unlike a chip encapsulant, however, the contact terminal is a thin layer of metal that is easily removed by mechanical scraping with a razor blade. Once the metal layer is removed, signal levels meet or exceed those found on the top, making bottom collection feasible and not overly invasive. Finally, the dual-probe top and bottom collection method is designed to lessen noise generated by the card itself, specifically that which is intentionally created as a side-channel attack countermeasure. This method is discussed in further detail in the following section.

3.9 Evaluation Technique

Results are obtained via measurement of real systems (instead of simulated data). Since the SCA technique relies on real-world leakage and manufacturing differences, simulation or modeling would be difficult and would provide unrealistic results of minimal utility. Additionally, SIM cards are readily available at low cost, interfacing specifications are well-documented, and the necessary equipment is on hand.

3.9.1 Component description.

3.9.1.1 SIM interface.

The first component of the measurement system is a custom SIM interface printed circuit board (PCB), shown in Figure 3.2. The PCB contains a SIM socket, a level-translating interface device, a high-precision crystal oscillator, and a clock buffer. The SIM socket provides simple mechanical and electrical connections to the SIM to hold it in

a steady, consistent position and to connect it to the rest of the circuit for communication and power. The level-translating chip buffers the SIM communication and translates it to the appropriate levels. Additionally, the chip provides SIM power-up sequencing, as SIMs require each pin to be activated in a sequential manner to begin communication. Finally, the crystal oscillator provides a stable and accurate clock signal to the SIM via the clock buffer. As collection data is analyzed for spectral content, the SIM must be clocked correctly at all times to ensure the accuracy of the observations. The SIM interface board is powered by a standard voltage-regulated laboratory power supply set to 3.3 volts.

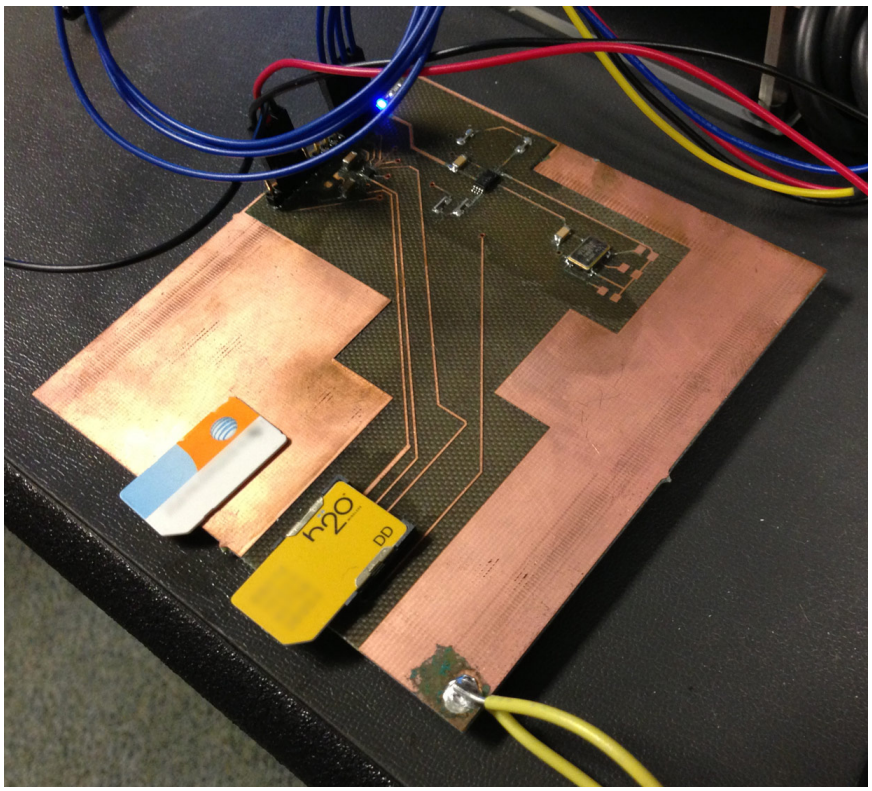

Figure 3.2: SIM interface board

3.9.1.2 FPGA controller.

Next, a Sasebo GII field-programmable gate array (FPGA) development board is used to perform SIM commands and provide a serial interface to the PC. The board features two

FPGAs, but only the Xilinx Virtex 5 chip is used in this work. The FPGA's logic (written originally in VHSIC Hardware Description Language (VHDL) code) receives commands from the PC and forwards them on to the SIM. In addition, the FPGA plays a critical timing role by generating a trigger signal to start and stop the capture of analog probe data. Since the exact time of PIN validation is unknown, the trigger must be wide enough (in the time domain) to guarantee that the capture period includes this comparison. Custom UART logic on the FPGA activates the trigger at the exact moment that the last bit of the PIN verification command is sent. As soon as the first bit of the SIM is received, the trigger is deactivated.

3.9.1.3 EM probe.

The next component is the data collection probe. A Riscure low-sensitivity (LS) EM probe is used to collect electromagnetic emanations from the SIM via electromagnetic coupling. Along with the collection coil, the probe contains a low-noise amplifier (LNA) to raise the level of the signal as close as possible to the source. The LNA is powered by a standard voltage-regulated laboratory power supply set to 5.0 volts. All collections are performed with the probe's shield in the down position to minimize noise from other sources (both from the SIM as well as environmental noise sources).

3.9.1.4 XYZ stage.

A Riscure XYZ stage is used to automate the precise positioning of the probe. The stage consists of a mounting block to hold an EM probe and a set of sliding rail mechanisms to give the probe three degrees of freedom. The movement of each rail is enabled by a stepper motor connected to a screw drive. A controller unit drives each of the motors and tracks their position such that the probe can be moved to an arbitrary 3D location based on commands sent to the controller's serial port. Typically, the X and Y axes are used to position the probe on the 2D surface plane of the SIM; the Z axis is set initially to achieve minimum probe-to-SIM clearance and is held constant for all experiments.

3.9.1.5 Oscilloscope.

A LeCroy WavePro 725Zi digital storage oscilloscope capable of 40 gigasamples per second measures and stores the analog signal from the electromagnetic probe. A second channel of the scope captures the external trigger signal used to initiate each data capture. The unit is controlled by the PC via an Ethernet connection to set capture parameters and download captured signal data. The oscilloscope is running the Microsoft Windows Vista Business SP1 operating system.

3.9.1.6 Collection and analysis PC.

Finally, a PC receives collected data from the oscilloscope, controls the XYZ stage, and communicates with the SIM via the FPGA board. The PC is also used to analyze the data post-capture. The PC is a Dell Precision T7500 workstation with two 2.0GHz Xeon E5504 processors and 48GB of RAM. All experiments are performed under the Microsoft Windows 7 Enterprise SP1 operating system. Relevant software running on the PC includes MATLAB version 2012b as well as Riscure Inspector version 4.4. MATLAB scripts perform all control operations for the SIM, oscilloscope, and XYZ stage. Inspector is used for manual data viewing and visual analysis. It also generates the signal strength plot from collected XY scan data.

3.9.2 Component connections.

The configuration of component interconnections is shown in Figure 3.3 and described below.

The SIM connects to the interface board via a socket which provides communication, a clock signal, a reset signal, and power. The SIM interface board connects to the FPGA board via wire leads for communication and signaling. The trigger signal connects to a standard oscilloscope probe, which in turn connects to the oscilloscope. The interface board is powered by a standard laboratory power supply.

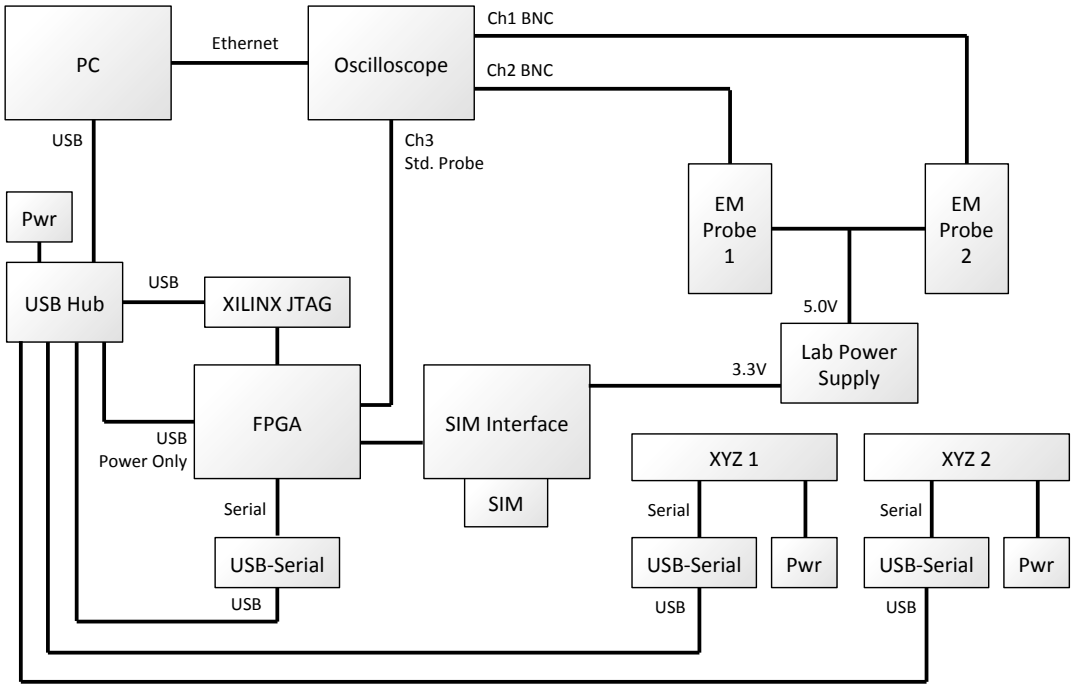

Figure 3.3: Component connections

The FPGA board is connected to a USB-to-serial adapter for communication, which in turn connects to a standard powered USB hub. A second USB cable is used to provide power to the FPGA. For programming, a Xilinx Platform Cable USB programming adapter is connected to the FPGA board's Joint Test Action Group (JTAG) port. The programming adapter then connects to the USB hub for communication.

Each XYZ stage connects to a USB-to-serial adapter for communication, which in turn connects to the USB hub. The stages each have their own 24V power supply.

Each Riscure EM probe connects to the oscilloscope via a Bayonet Neill-Concelman (BNC) cable. Filters are connected in-line between the BNC cable and the oscilloscope. The probes each connect to the laboratory power supply.

The oscilloscope receives the signals from the two probes on channels 1 and 2. The trigger signal received from the FPGA via a probe is connected to channel 3. The oscilloscope connects to the PC via a direct Ethernet connection.

The laboratory power supply is connected to each probe's LNA (5.0V) as well as the SIM interface board (3.3V).

The PC is connected to the USB hub for communication with the FPGA, XYZ stages, and the FPGA programming adapter. It is also connected directly to the oscilloscope via Ethernet.

3.9.3 Experimental process.

The experimental process follows the approach in Section 3.2.2 closely. First, a test SIM is chosen and exploratory data analysis is performed by executing an XY scan of the device. This scan records the signal strength at each location in a grid of locations across the die surface of the SIM. Typical scan dimensions are 20x20 or 30x30, resulting in 400 and 900 collections, respectively. As the probe moves to each location, a known-good PIN is validated and the oscilloscope is triggered to capture the data. When the capture is complete, the Inspector software is used to display the scan results in the form of a representative signal strength level at each location. The optimum capture location is determined visually from analysis of these results while adjusting a software bandpass filter to hone in on the expected frequency. When the capture location is chosen, its coordinates are recorded and the XYZ stage moves the probe to this location.

The training phase now begins by manually setting the SIM to a fixed PIN. Next, the automated process begins as controlled by MATLAB. A large, random selection of invalid PIN codes are then presented to the card, alternating each time with the (known) correct PIN to reset the retry counter. The probe and oscilloscope capture each invalid attempt as triggered by the FPGA. A MATLAB script requests each capture's data from the oscilloscope and saves it to a local file on the PC. Each individual capture record also

stores the attempted PIN. After initial visual inspection for alignment, MATLAB scripts process the data in a manner specific to the chosen correlation algorithm.

For the PIN recovery phase, a capture technique is chosen and the SIM is set to an "unknown" PIN. The attempt-and-collect process is repeated $n - 1$ times, where n is the number of attempts remaining on the retry counter. A random PIN is used each time, and the retry counter is not reset, since the correct PIN is "unknown." Once the last retry value is reached, the correlation algorithm computes the most likely correct PIN based on the collected data. Finally, this guessed PIN is sent to the SIM for validation and the result recorded. Results are validated by the SIM's response to the validation command. Alternately, to prevent accidental card lock-out, this final validation step may be performed by comparing the guessed PIN to the recorded "unknown" PIN rather than sending it to the SIM for verification. This method of validation is equally effective as no electromagnetic data collection is performed during the final validation – the goal is simply to determine the correctness of the guessed PIN.

3.10 Experimental Design

Given the selected factors listed in Section 3.8, a full factorial design is feasible and gives the best representation of the effects of changing factors. There are two factors with three levels specified for each, giving $3 \times 3 = 9$ experimental configurations. Each experiment is repeated five times to establish results independent of the noisy electromagnetic environment. This gives a total of $9 \times 5 = 45$ experiments.

3.11 Methodology Summary

A methodology is presented to identify, capture, and exploit electromagnetic emanations of SIM cards during PIN verification. The process consists of a training or learning phase as well as an execution phase. The process is preceded by exploratory data analysis to ensure the appropriate leakage is found. Appropriate factors are varied

to examine the results of differing trace counts and collection methods. An experimental equipment setup is described to control the SIM and perform the necessary collection and analysis. Finally, a full-factorial experimental design is proposed resulting in a total of 45 experiments.

IV. Results

4.1 Introduction

This chapter presents and analyzes the results of executing the methodology described in Chapter 3. Section 4.2 discusses assumptions made and parameters held at fixed values throughout the experiments. Section 4.3 describes the output of the experiments. Section 4.4 presents an analysis of these results and summarizes the outcome.

4.2 Fixed Values and Assumptions

Before the discussing experimental results, it is important to note a number of assumptions made and fixed values chosen.

4.2.1 SIM selection.

Initially, the intent was to test a variety of SIM makes and models against this methodology. Cards from Gemalto, Oberthur, Infineon, and STMicroelectronics were obtained and tested. However, limitations in the custom UART design, variations in SIM communication protocols and timing, and the author's limited experience in VHDL all contributed to difficulty in reliably interfacing with multiple models of SIMs. As such, a single representative SIM was chosen and development effort was focused on obtaining reliable and consistent communication with that particular SIM.

The SIM chosen is an AT&T-branded standard-size (2FF) SIM. The AT&T branding is chosen due to the company being the largest GSM-based network provider in the United States, thus, representing one of the greatest SIM card distributions in the country. The SIM is readily available at low cost (around $3.00) and is compatible with GSM, UMTS, and LTE networks. Available since at least 2010, the security features, manufacturing process, and architecture of this SIM are thought to be representative of most modern SIMs.

Unfortunately, information on the manufacturer, model, and chipset information is difficult to locate for most SIMs, and cards are often labeled only with a carrier part number. AT&T does not publish the specifications for this particular part number. However, multiple unofficial sources associate the part number and unique contact pad layout with Infineon (the chipset/CPU manufacturer) and Gemalto (the integrator and card manufacturer) [21], [20], [17]. Thus, SCA countermeasures known to be used by Infineon are thought to be present in this device.

4.2.2 Clock assumptions.

As discussed earlier, it is necessary to filter the input signal to lessen electromagnetic noise picked up by the EM probe and any components between the probe coil and the oscilloscope (including the cable, LNA, etc.). The source clock signal provided to the SIM by the interface board is a 4 MHz square wave, so the SIM CPU is first assumed to operate at a clock rate of 4 MHz. The chosen filter must allow this frequency to pass; as an 11 MHz low-pass filter is the closest value found on-hand, it is selected and affixed.

A front-side (top-of-die) collection is taken starting with the SIM at idle and continuing through a PIN verification command. The resulting signal is shown in Figure 4.1.

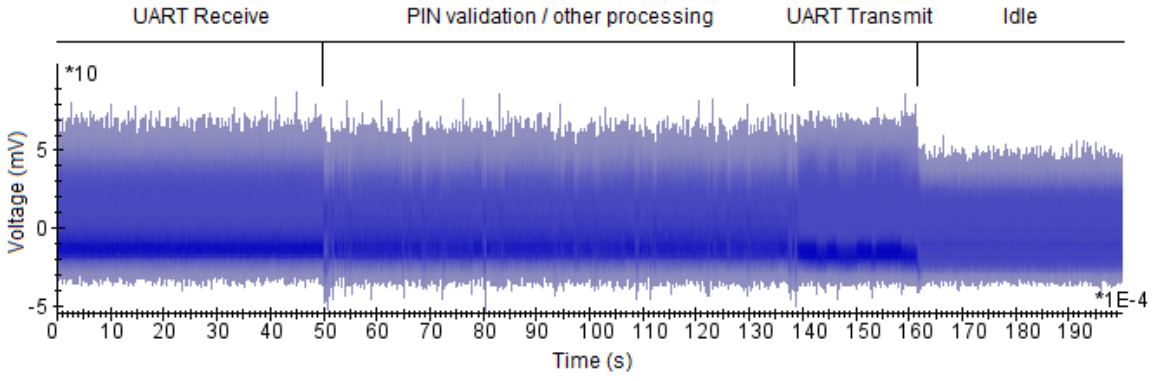

Figure 4.1: Macro-level trace overview (front side)

The structured nature of the signal and the lower levels observed at idle are good indications that the captured signal is associated with CPU activity. A power spectral density (PSD) analysis of the same collection (Figure 4.2) shows that the strongest frequency component of the signal is located at 4 MHz, which further supports that conjecture.

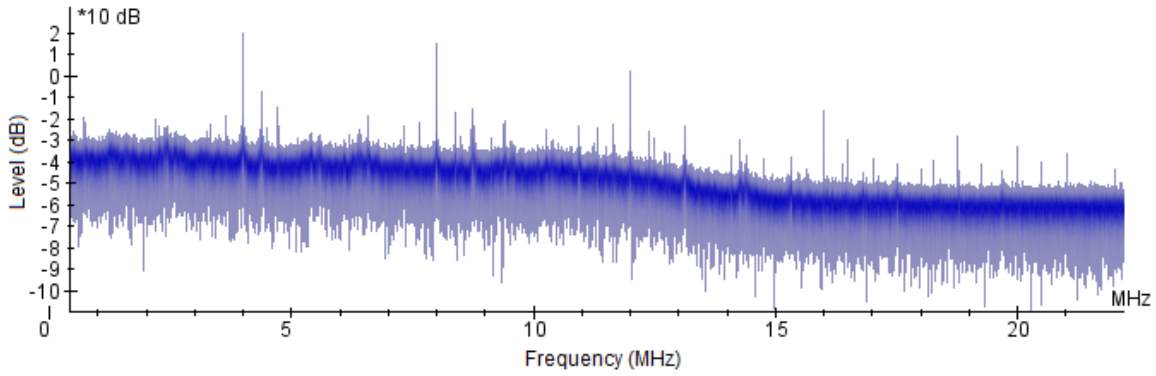

Figure 4.2: Unnormalized PSD analysis

The oscilloscope's sampling rate must be configured to accurately capture data at these frequencies. While the selected oscilloscope is capable of 40 gigasamples per second (GS/s), capturing at that rate produces an inordinate amount of unnecessarily detailed data, increasing processing time and storage requirements. Instead, the sampling rate is reduced by an order of magnitude into the megasamples per second (MS/s) range. If the input signal to the scope is band-limited to 11 MHz (f), the Nyquist sampling theorem suggests that sampling can theoretically be performed at 22 MS/s ($2f$, the Nyquist rate) without losing sinusoidal frequency information. However, to avoid aliasing, the sampling rate is arbitrarily chosen to be double the Nyquist rate, or 44 MS/s. The closest sampling rate settings the oscilloscope provides are 25 MS/s and 50 MS/s, so the higher rate is chosen and the sampling rate is fixed at $fs = 50$ MS/s. Note that this rate is over six times the Nyquist rate of the signal of interest ($f = 4$ MHz, Nyquist $= 2f = 8$ MS/s).

4.3 Experimental Output

4.3.1 XY surface PSD scans.

The result of an XY surface scan of the device follows in Figure 4.3. Each plot shows the average PSD energy level for the selected frequency band at each physical location on the die. Note that each plot is dynamically scaled (normalized) and thus direct intensity comparisons cannot be made between plots. Scan dimensions are 30x30 measurements, with the extrema being defined by the locations where the probe is centered on the respective die edge. The trigger is timed from the FPGA's transmission of the last bit of the PIN verification command to the detection of the first bit of the response returning from the SIM.

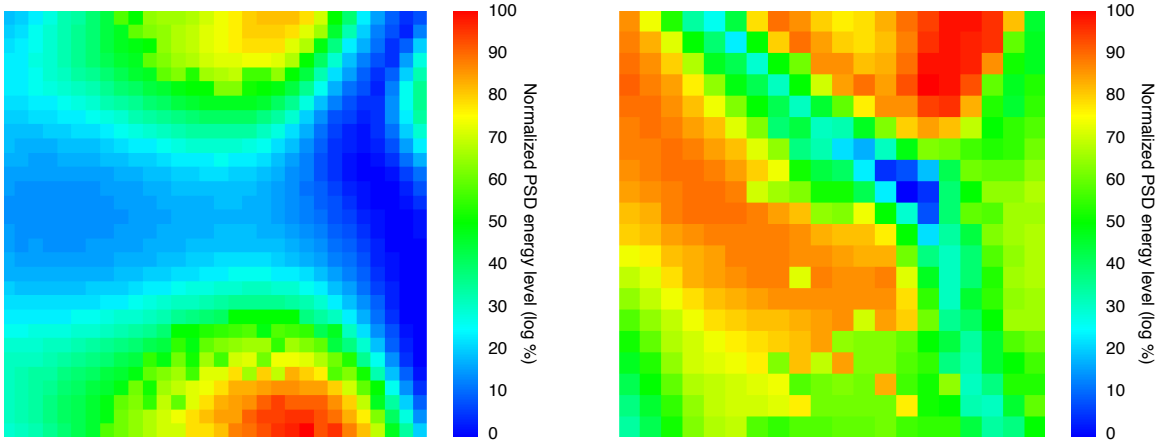

Figure 4.3: Initial unfiltered XY scan of SIM front (left plot) and back (right plot).

After applying a narrow-band software filter (in addition to the 11 MHz low-pass hardware filter) to only pass the desired 4 MHz signal, the XY plots appear as shown in Figure 4.4. On each side, the location with the strongest signal level is chosen as the capture location.

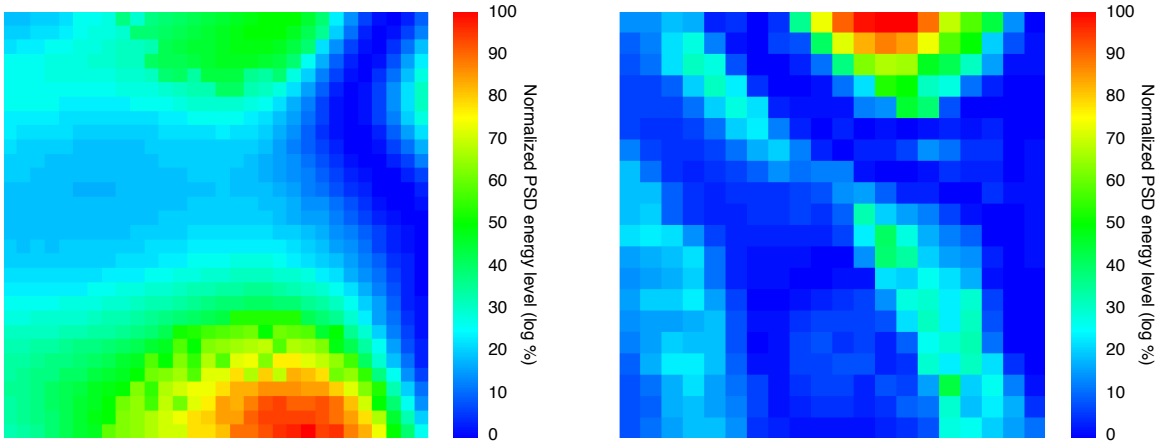

Figure 4.4: 4 MHz filtered XY scan of front (left plot) and back (right plot).

4.3.2 Initial analysis.

Next, collections are taken from each probe at the 25,000 trace level. The highest level is chosen as the high number of traces, while slower to work with, are likely to provide the strongest correlation due to the sheer amount of data. These collections could have been performed simultaneously as the test setup allowed the front and back probes to move independent of each other and the SIM location. However, for the sake of development time, front and back collections are taken independently. Results shown are from back-side collections; front-side collection results are essentially identical but contain more noise.

Initial analysis shows the macro-level operation of the PIN verification (shown in Figure 4.5) is as follows.

Immediately, the instruction dependency is visible. The high-level functionality is very apparent and is consistent among each repetition. First, the trailing UART communication from the FPGA to the SIM is seen. Next, the verification process (the details of which are unknown) is apparently repeated three times. Each repetition is approximately 3 milliseconds (ms) long. Following the three repeated processes, another UART communication period is seen, this time from the SIM to the FPGA (the SIM's

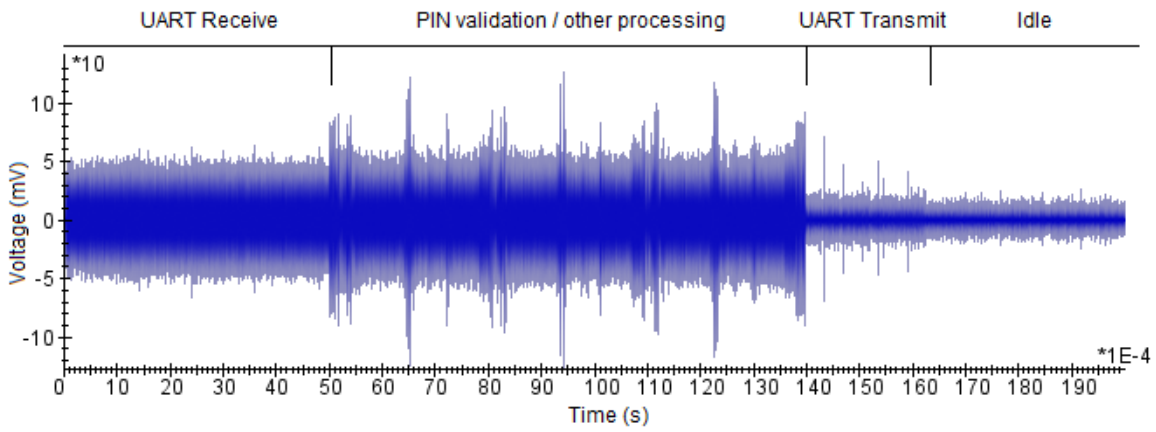

Figure 4.5: Macro-level trace overview (back side)

response to the PIN validation). Afterwards, the SIM enters what appears to be an idle state, where it remains for the rest of the capture period.

The trigger signal (not shown) from the FPGA to the oscilloscope is fired at approximately the 5 ms mark and is deactivated at 14 ms. This 9 ms period is the entire time between command and response, so the PIN verification must take place inside this window. If the SIM is presumed to be running at a clock rate of 4 MHz, this window contains approximately 36,000 clock cycles.

Observing the signal's PSD plot in Figure 4.2, the source clock signal is visible as a 4 MHz pulse along with several of its harmonics at 8 MHz and so on. This provides a "sanity check" that the emanations of the SIM are being captured. A spectral analysis is also conducted with Inspector, which uses the PSD (calculated from the discrete Fourier transform as $|\text{DFT}|^2$) to display the signal's representation in the frequency domain. Here, peaks are also present at 4 MHz and subsequent harmonics. There are various smaller peaks but the dominating frequency is the 4 MHz clock. The rolloff of the 11 MHz filter is also apparent.

Observation at this 4 MHz level shows that clock pulses are reasonably well aligned at this point. Unfortunately, some of the larger features are not, a potential side effect

of capturing such a large (and long) window. It does not appear that random clocking is occurring, as the observed clock signal maintains alignment throughout the different traces.

4.3.3 Standard attacks.

Before any attacks are performed, the SIM's PIN is manually set to the value 1111. All tests are conducted with the value 111x, that is, the first four digits are correct while the fourth digit is varied randomly. As discussed in the methodology, the key value used is attached to the metadata of each trace so that it can be used as the plaintext input to the DPA attack later on.

An initial analysis is conducted by partitioning the data by the incorrect key guess. This results in ten groups based on the value of the fourth digit of the attempted PIN (0-9). A visual comparison of traces is performed to look for similarities. While the same macro structure is observed among the traces, many smaller areas appear to vary substantially even when the attempted PIN is the same. Figure 4.6 shows a zoomed area of interest in a singe trace for which the PIN guess is 1110. Figure 4.7 shows a composite of 100 traces of the same PIN guess (1110) zoomed to the same area as in Figure 4.6. These two traces are aligned temporally (based on the signal increase when processing begins), but the identifiable "spike" changes in location and intensity with each collection, as do the smaller features closer to the noise floor.

Next, a difference of means test is conducted to attempt to cancel out some of the noise. Each PIN value group contains approximately 2,500 traces, at which point random noise should approach an average of zero effect. As an example comparison, the mean traces of PIN attempts 1110 and 1117 are shown in Figure 4.8. These values were particularly chosen as the Hamming weights are the furthest apart (the last digit of 1110, encoded as 0x30, has a Hamming weight of 2; the last digit of 1117, encoded as 0x37, has a Hamming weight of 5). Unfortunately, the differences are not consistent, and subsequent collections rank the different PIN groups differently. Figure 4.8 shows a representative zoomed region

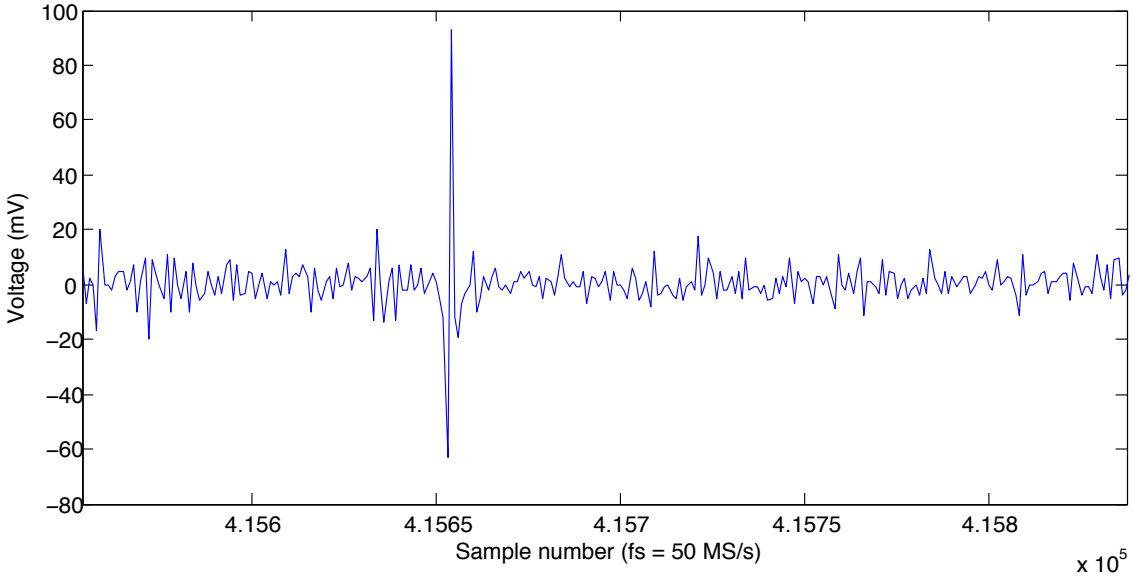

Figure 4.6: Zoomed section of trace with PIN guess 1110

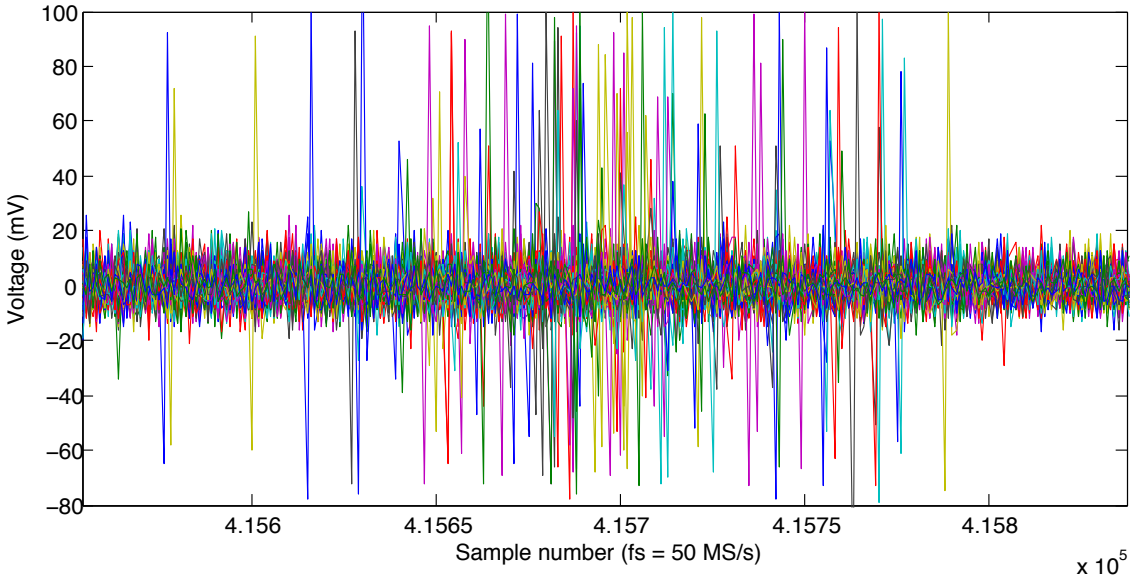

Figure 4.7: Zoomed composite section of 100 traces with PIN guess 1110

of the inconsistent and noisy result – very little correlation is visible. Plots comparing the remaining PIN guesses appear similarly inconsistent.

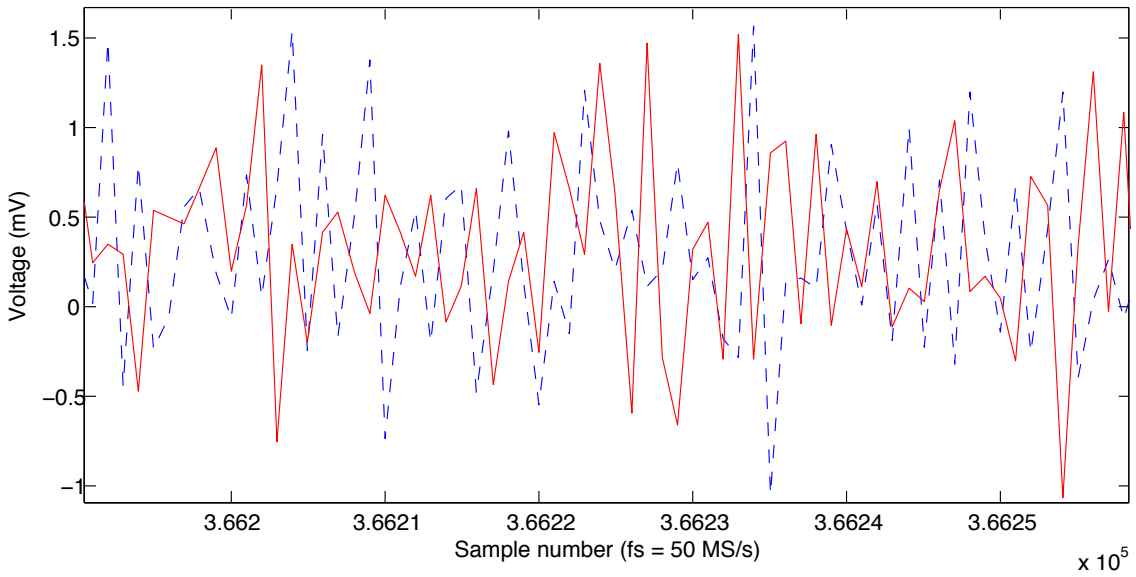

Figure 4.8: Comparison of means, 200 traces: PIN guess 1110 and PIN guess 1117

Although visual analysis did not reveal any obvious details, further analysis is conducted to ensure data is not hiding below what is visible by the naked eye. Simple electromagnetic analysis is the first attack attempted. First, all traces are compared via a standard deviation plot. If SPA is viable, an obvious instruction-level correlation should exist, and differences should "pop out." This plot comparing all traces is shown in Figure 4.9. Additionally, the average of many traces with a given PIN should not differ substantially from any single trace. This is shown by plotting the standard deviation of only traces with a single PIN guess (1110) as shown in Figure 4.10.

Substantial differences appear in both plots, and the peaks in the same locations indicate little difference in signal variation as a result of a data dependency. A fair amount of noise is present as well. The standard deviation of the noise present in the idle state is measured at approximately 35 millivolts (mV) peak-to-peak and disappears when the SIM is removed from the measurement apparatus, which indicates that the SIM itself is the source of this noise. Possible sources for this noise include the SIM's internal power

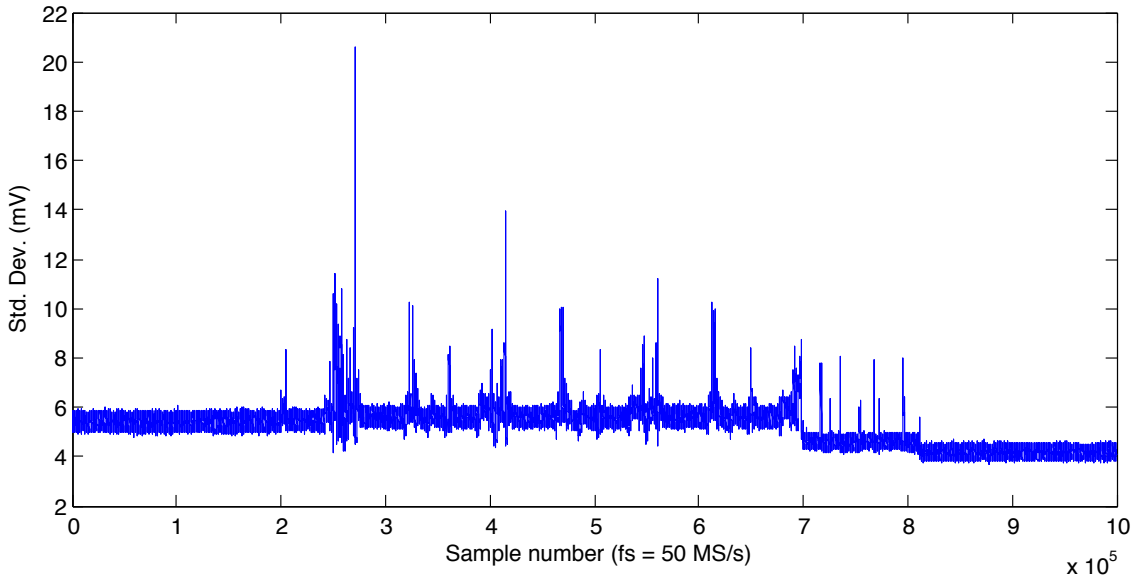

Figure 4.9: Standard deviation, 1000 traces: All PIN guesses 1110-1119

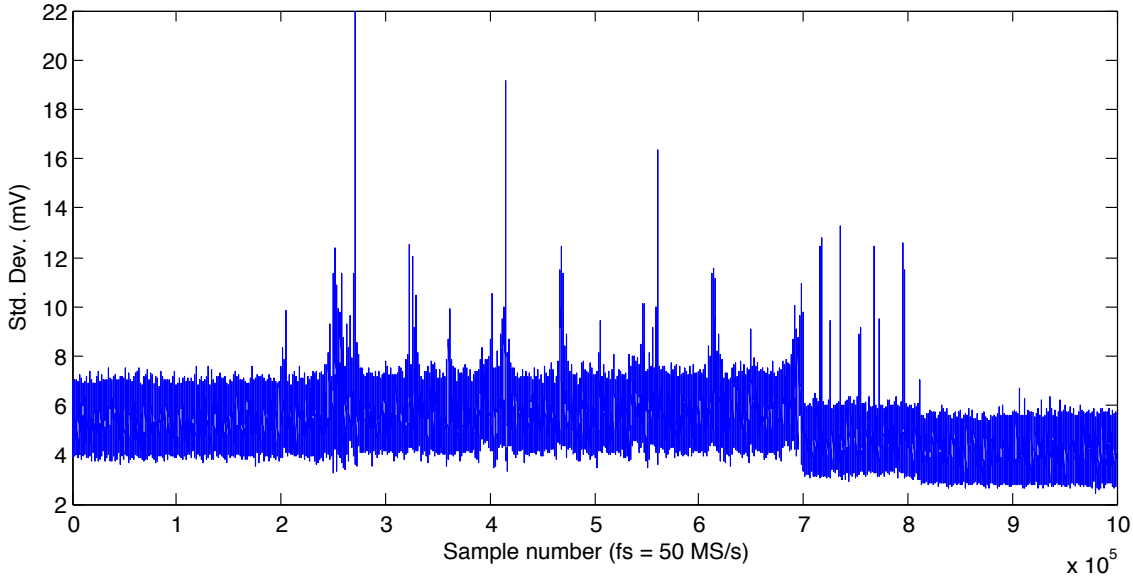

Figure 4.10: Standard deviation, 100 traces: PIN guess 1110

supply, clock manager, or routine operating system tasks. The noise could also be coupled

from the external power source. Alternately, the source could be a noise generator present on the SIM as an SCA countermeasure.

Next, a DPA attack is attempted. Unfortunately, DPA requires a model in which an intermediate state for a given plaintext is known. In this case, encryption is not explicitly taking place, although it is likely that encryption or hashing is occurring internally to compare the input PIN to a stored encrypted/hashed PIN. However, no knowledge of this algorithm is given, so the attack is conducted with the assumption that a standard comparison is used. A common instruction used by processors to perform a comparison is to exclusive OR (XOR) two values together, so XOR is chosen as the operator for the DPA function. The collected 25,000 trace set is used and a DPA attack is performed using key guesses 0 through 9 (encoded as 0x30 through 0x39), as it is known that these are the only plaintext values used.

The DPA guess correlation plot for the fourth (varied) byte is shown in Figure 4.11. Unfortunately, the plot does not provide promising results. There are no significant peaks, and even if the maximum peak value is selected, the chosen peak is not substantially greater than the noise level. The correlation value never exceeds 0.15 (a value of 1 indicates perfect correlation). Subsequent runs of the collection process (with data captured in the exact same manner) produce different key guesses every time.

If correlation was found for a given key guess, a large correlation spike would be shown at the key's location in the plot, along with possibly a few false spikes. However, no obvious outliers exist here as no correlation level is substantially above the "noise floor" of the curve. There are a number of reasons this may have failed, but the most likely reason is the lack of architectural knowledge. This test assumes a simple XOR is used to perform the PIN comparison, but PIN encryption or a more obfuscated comparison could completely invalidate that hypothesis with trivial effort. Further hypotheses could be tested blindly, but more analysis to determine any details of the algorithm in use is a stronger approach.

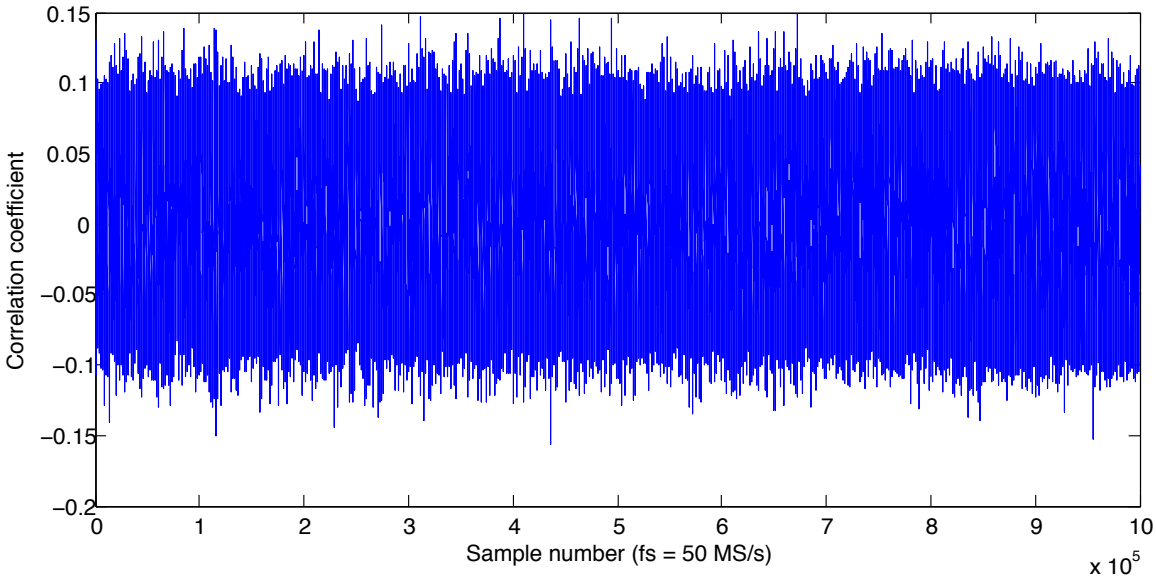

Figure 4.11: Correlation plot for DPA (modified byte)

4.3.4 Noise cancellation.

As noise (either from a lack of leakage or due to SCA countermeasures) appears to be a problem, a novel dual-probe approach is presented as a means of removing the noise. This process is based on a noise cancellation technique used in the field of audio production and processing. In an environment containing large amounts of background noise, two microphones are often used to function as a noise-canceling pair. One microphone captures the subject speaking along with the background noise, while the other microphone is turned in order to capture only the noise. The noise signal is then subtracted from the speaker's voice signal, resulting in cleaner noise-reduced audio.

This principle is used here in electromagnetic analysis application. First, the assumption is made that the active noise generator device contributes noise to the system in an additive fashion. System leakage A is corrupted by noise source B to produce collected EM signal C, trivially represented $A + B = C$. A single probe detects C and captures it, but the corruption due to noise makes the signal unusable for EMA.

Now, suppose that noise source B is collected separately by a second probe. If the cross-sectional area of the probe coil is small enough to collect only the generated noise and not the surrounding signals, C can then be processed by subtracting B in post-processing or in real-time by the oscilloscope. The resulting signal $C - B$ is the original A, no longer corrupted by noise. The key to success is finding the location where noise is generated, and capturing that signal in as isolated a fashion as possible.

To accomplish this, the second probe is positioned on the opposite side of the device from the first probe (i.e., pointed towards top surface of the die instead of the bottom surface) so as to provide maximum flexibility in locating the noise source. An XY scan is conducted to locate sources of noise generation during the desired operation. Depending on noise generator implementation, this source could be located in a separate circuit on the die, or it could be found in the active shield's connection to the rest of the circuit. Either way, once the noise source is located, it must be recorded or added to the source signal in real time. Note that if the same model probe is used as on the top side of the die, the signal is already inverted with respect to the result C (as the physical inversion of the probe causes the coil to be wound in the opposite direction relative to the other probe) and must be added instead of subtracted.

In this particular implementation, this dual-probe method did not succeed. Skew was a constant issue as it was difficult to get the two signals temporally calibrated together, even when a reference clock signal was used for initial calibration. Slight movements of the probe caused the signals to drift out of synchronization once again, making the use of this method in an attack impractical for the time being. A sample well-aligned trace is shown in Figure 4.12 (the "Difference" trace is the result).

Despite the lack of complete success, the subtraction does appear to lessen noise in the source signal. This method holds promise for future research and should be studied under more carefully controlled conditions and on additional SIM and smart card configurations.

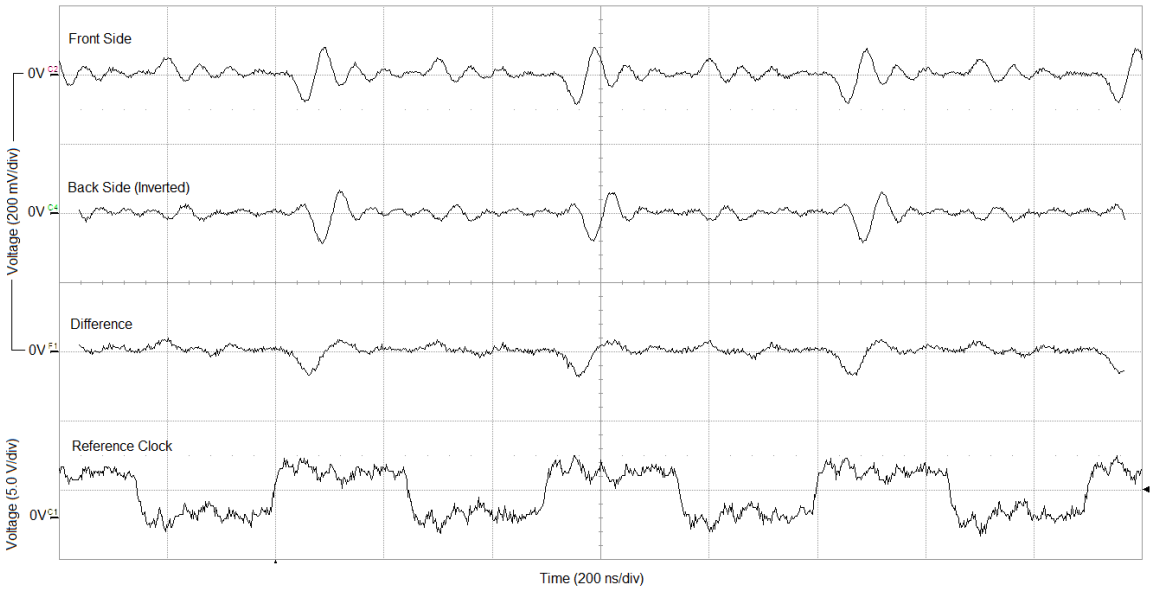

Figure 4.12: Illustration of background noise canceling process

4.4 Result Analysis

All experiments shown above are repeated at least five times for each of the different trace count levels as discussed in the methodology. As expected, capturing fewer traces increases the noise level and the resulting data has less utility than the 25,000-trace collections shown above. EMA is shown to be ineffective for PIN recovery given the selected parameters and assumptions, but future work can vary these values further and may result in more information leakage. Unfortunately, these results are inconclusive. However, while the EM data does not exhibit a strong data dependency, the SIM does show a strong operation dependency, that is, the macro level operation of the PIN validation is clearly and consistently visible.

V. Conclusions and Future Work

This chapter presents an overall summary of the research conclusions, lessons learned, and future work opportunities. Section 5.1 summarizes the analysis of the results described in Chapter 4. Section 5.2 considers lessons learned throughout this research. Finally, Section 5.3 describes future research opportunities based on this work.

5.1 Research Conclusions

The results of using electromagnetic (EM) side-channel analysis to observe a SIM card PIN are inconclusive. A simple electromagnetic analysis attack presented too much noise to identify any meaningful differences resulting from different PIN codes. A naïve attempt at differential analysis operating on the same data produced similar results, but the lack of architectural knowledge prevented the use of an informed attack on whatever encryption algorithm is used.

One of the largest factors contributing to the result was a lack of understanding of the PIN validation algorithm. Unlike encryption algorithms, which are well-known, documented, and can be studied outside of the device, the PIN validation algorithm used by any given card is completely unknown and is truly a "black box." In addition, the three- or ten-attempt limits greatly hinder attacks that typically require tens if not hundreds of traces.

Another factor is the lack of countermeasure information. In every card studied (prior to limiting the EMA attacks to a single card), the type of countermeasures employed was unknown. Mere knowledge of the existence of a countermeasure (without implementation details) would be enough to devise a workaround, but without even that information, it is difficult to know what to look for. For example, if the card was known to insert random

NOP instructions into the execution pipeline, these could be identified and removed in post-processing of the data.

While no data-dependent leakage was found, the card's EM response clearly indicated that a PIN code was being evaluated, as the EM signature of the validation operation was consistently visible to the naked eye after only a single trace was taken. It is possible that this information could lead to other attacks on SIMs and other smart cards, since at least in this case, the internal operation was not completely hidden.

The results of this research have little standalone impact, but the methodology, code, circuits and results can be used in future work to further evaluate the concepts studied here.

5.2 Lessons Learned

Many lessons about black-box analysis were learned over the course of conducting this research. There were many times in the process of this research where much time was lost by following a path based on an assumption that later turned out to be false. One of these assumptions was made in the selection of a filter frequency. Early literature review indicated that the SIM would operate at the supplied clock frequency internally; thus, a relatively slow (4 MHz) clock was chosen and an 11 MHz low-pass filter installed into the collection apparatus. It was not until the end of the research that this initial choice was reconsidered and further clock frequencies were discovered higher in the spectrum. Due to time constraints, a thorough analysis has not been conducted on these higher frequencies; however, with what is now known about the SIM's use of a PLL to obtain a higher CPU frequency, it seems likely that greater information leakage would be found in these higher bands.

Another lesson learned involved the use of side-channel analysis equipment. As the probe equipment present in the laboratory was well-known to be of high quality, the use of other probes was not considered. The probe coil only has a cross-sectional area of 1 mm^2, which was unable to provide a very detailed image of the tiny $< 4\,mm^2$ die. Despite

this limitation and several time-consuming equipment-related problems with the probe, research into another type of probe was never conducted. Recent research indicates that a probe with much greater resolution can be constructed in an afternoon out of simple parts readily available in the laboratory. Had this possibility been considered earlier, less time would have been spent attempting to work around the probe limitations.

5.3 Future Work

Ideas for future work fall into two categories: those that continue this methodology of extracting the PIN from a SIM, and those that may be more generally useful in side-channel-related work.

5.3.1 SIM PIN work.

- *Explore other frequencies.* An assumption made from the beginning of the research was that the SIM's internal CPU clock was relatively low; thus, the input signal was filtered as required to hone in on this signal. After removing the filter and observing the spectrum in a fast Fourier transform (FFT) waterfall plot, several other frequencies appear that are good candidates for the CPU frequency. Further research should explore these frequencies and attempt the same EMA attacks at the most likely spectral location. Additionally, the XY scan could be modified to show spectral content at each physical location to better identify the different components (PLL, power supply, CPU, etc.).

- *High-resolution probe.* As mentioned in Section 5.2, the resolution of the probe used in this research (1 mm^2) is very low compared to the SIM's die size. Deutschmann in [9] demonstrates a means of easily constructing high-resolution probes with very impressive results (200μm and 50μm). A higher resolution probe would not only allow more powerful focusing on key leakage locations, but it could also improve the accuracy of the proposed dual-probe noise reduction technique.

- *Sinusoidal clock.* Both the frequency- and time-domain plots of the electromagnetic trace show many harmonics of the 4 MHz reference clock signal. Research suggests that using a sinusoidal clock input rather than a square wave would likely cut down on those harmonics and present a cleaner and more easily analyzed signal.

- *Additional card makes/models.* The make and model of SIM in this research was limited to a single selection due to time and VHDL ability. Future research should update the VHDL code to properly communicate with additional cards; if communication with these other cards is made operational, it is possible that this methodology would function as-is and perhaps even succeed with another SIM. Additionally, many other smart-card-based authentication systems use a PIN code for access and are potentially vulnerable to this type of attack.

- *Baseline testing.* Existing research on SIM PIN extraction developed code and ran it on live SIMs for analysis [18] [14]. Future research should obtain a development kit for a modern SIM equipped with countermeasures, develop SIM verification code on it, and implement a working side-channel attack. This would result in a baseline study of the effects the countermeasures actually have in the real world, and this information could be used to strengthen the attack on existing SIMs.

- *Counterfeit/malware detection.* The SIM card tested in this research exhibits a consistent pattern of electromagnetic leakage during the PIN verification process. This pattern is consistent across multiple traces and is likely dependent on the operations running on the SIM's CPU. These patterns could be used as a signature to ensure that a given SIM does not contain counterfeit or malicious code.

- *Advanced XY scan.* Current techniques to scan for EM emissions only operate in a single plane. It may be possible to detect additional signals if the card is examined

in additional planes (e.g., the sides of the card) or from alternate angles by attaching the EM probe to an articulated robotic arm.

5.3.2 Related SCA work.

- *Dual-probe approach.* The dual-probe noise canceling approach discussed in this research appears to be novel in this SCA application. Further analysis of the dual probe approach is warranted and may be useful to reduce the effect of more aggressive noise generation systems. Baseline testing of this method's efficacy should be performed against a known algorithm running on a SIM architecture known to employ noise generation countermeasures.

- *FM demodulation.* Analysis of the FFT waterfall plot of the SIM's spectral output during PIN validation reveals an interesting signal in the 34 MHz range. The signal varies in frequency over time, covering an approximately 1.5 MHz band. The macro-level variation pattern seems consistent across multiple traces. One potential use for this signal could be to correct a corresponding time-domain EM trace for clock variation countermeasures. It may also be a side-channel leakage itself. An interesting future research study would be to take a large collection, demodulate this signal to convert it to the time domain, then apply standard DPA-type analysis to the data.

5.4 Summary

In conclusion, the study on the side-channel susceptibility of SIM card PIN validation routines to side-channel analysis attacks is inconclusive. This research demonstrates that naïve attempts at simple and differential electromagnetic analysis do not reveal the PIN of the tested SIM, but it is unclear whether this is the direct result of intentional hardware or software countermeasures or simply a side-effect of the validation algorithm's implementation. Despite the lack of PIN leakage, the SIM's EM emanations do contain an

instruction dependency for the PIN validation. Recommendations include exploring higher frequency bands and performing baseline testing with known SIM architectures. This research provides results of known analysis techniques as well as a thorough methodology for further study of additional SIMs and collection methods.

Bibliography

[1] Aigner, Manfred and Elisabeth Oswald. "Power analysis tutorial". *Institute for Applied Information Processing and Communication, University of Technology Graz-Seminar, Tech. Rep*. 2000.

[2] Alliance, Smart Card. *What makes a smart card secure?* Smart Card Alliance, 2008.

[3] ATT Developer Program. "Network Technologies - Authentication and Encryption", 2012. URL http://developer.att.com/developer/forward.jsp?passedItemId=2400370.

[4] Barbara, John. "SIM Forensics: Part 3", 2011. URL http://www.dfinews.com/article/sim-forensics-part-3.

[5] Bezakova, Ivona, Oleg Pashko, and Dinoj Surendran. "Smart Card Technology and Security", 2000. URL http://people.cs.uchicago.edu/~dinoj/smartcard/security.html.

[6] Briceno, Marc, Ian Goldberg, and David Wagner. "A pedagogical implementation of A5/1", 1999. URL http://www.scard.org/gsm/a51.html.

[7] Dacs and WhiteTimberwolf. "SmartCardPinout.svg", 2009. URL http://commons.wikimedia.org/wiki/File:SmartCardPinout.svg.

[8] De Mulder, E., P. Buysschaert, S.B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede. "Electromagnetic Analysis Attack on an FPGA Implementation of an Elliptic Curve Cryptosystem". *Computer as a Tool, 2005. EUROCON 2005.The International Conference on*, volume 2, 1879–1882. 2005.

[9] Deutschmann, B. and R. Jungreithmair. "Visualizing the electromagnetic emission at the surface of ICs". *Electromagnetic Compatibility, 2003. EMC '03. 2003 IEEE International Symposium on*, volume 2, 1125–1128 Vol.2. 2003.

[10] Dhem, J.F., F. Koeune, P.A. Leroux, P. Mestré, J.J. Quisquater, and J.L. Willems. "A practical implementation of the timing attack". *Smart Card Research and Applications*, 167–182. Springer, 2000.

[11] European Telecommunications Standards Institute. *ETSI TS 102 241 V7.9.0. Smart cards; UICC Application Programming Interface (UICC API) for Java Card (Release 7)*. European Telecommunications Standards Institute, Jun 2008.

[12] European Telecommunications Standards Institute. "3GPP Confidentiality and Integrity Algorithms", 2012. URL http://www.etsi.org/services/security-algorithms/3gpp-algorithms.

[13] European Telecommunications Standards Institute. *ETSI TS 121 111 V11.0.1. Universal Mobile Telecommunications System (UMTS); LTE; USIM and IC card requirements*. European Telecommunications Standards Institute, 2013.

[14] Ferenc, Jakub. "Power analysis of the PIN verification procedure on smart cards (English title, translated)", 2006 [cit. 2013-05-13]. URL http://is.muni.cz/th/98993/fi_b/.

[15] Flylogic Engineering. "ST201: ST16601 Smartcard Teardown", 2007. URL http://www.flylogic.net/blog/?p=18.

[16] Flylogic Engineering. "Infineon/ST Mesh Comparison", 2010. URL http://www.flylogic.net/blog/?p=86.

[17] Flylogic Engineering. "(No title, Twitter communication)", 2010. URL https://twitter.com/semiconduktor/status/23965552475.

[18] Folkman, Lukáš. "The use of a power analysis for influencing PIN verification on cryptographic smart card", 2007 [cit. 2013-05-13]. URL http://is.muni.cz/th/140414/fi_b/.

[19] Gandolfi, Karine, Christophe Mourtel, and Francis Olivier. "Electromagnetic Analysis: Concrete Results". etinK. Ko, David Naccache, and Christof Paar (editors), *Cryptographic Hardware and Embedded Systems CHES 2001*, volume 2162 of *Lecture Notes in Computer Science*, 251–261. Springer Berlin Heidelberg, 2001. ISBN 978-3-540-42521-2. URL http://dx.doi.org/10.1007/3-540-44709-1_21.

[20] Gemalto NV. "GemXpresso R4 E36/E72 PK ... Security Policy", 2009. URL http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140sp/140sp771.pdf.

[21] Harber, David. "Gemalto Test Card Procurement Process", 2010. URL http://ekb.spirent.com/resources/sites/SPIRENT/content/live/FAQS/10000/FAQ10956/en_US/16077%20-%20Gemalto%20Test%20Card%20Procurement%20Process%20v1_9.pdf.

[22] Huang, Andrew. *Hacking the Xbox*. No Starch Press, Inc, San Francisco, CA, 2003. ISBN 1-59327-029-1.

[23] Infineon Technologies AG. "SLE 66CLX360PE(M) Short Product Information", 2006. URL http://www.infineon.com/dgdl/SPI_SLE66CLX360PE_1106.pdf?folderId=db3a304412b407950112b408e8c90004&fileId=db3a304412b407950112b4099d6c030a&location=Search.SPI_SLE66CLX360PE_1106.pdf.

[24] Infineon Technologies AG. "SLE 76CF3601P Short Product Information", Mar 2007. URL http://www.infineon.com/cms/en/product/chip-card-and-security-ics/security-controller-contact-based/sle-76-family:

-flash-controller-for-sim-uicc-applications/channel.html?channel=db3a3043156fd57301161520ab8b1c4c.

[25] Infineon Technologies AG. "Security controller (contact-based)", 2013. URL http://www.infineon.com/cms/en/product/chip-card-and-security-ics/security-controller-contact-based/channel.html?channel=ff80808112ab681d0112ab6929fc0138.

[26] ISO/IEC. *ISO/IEC 7816. Identification cards – Integrated circuit cards*. ISO/IEC, 2011.

[27] Jansen, Wayne and Rick Ayers. "Forensic Software Tools for Cell Phone Subscriber Identity Modules", 2006. URL http://csrc.nist.gov/groups/SNS/mobile_security/documents/mobile_forensics/pp-SIM-tools-final.pdf.

[28] Jansen, Wayne and Aurelien Delaitre. "Reference Material for Assessing Forensic SIM Tools", 2007. URL http://csrc.nist.gov/groups/SNS/mobile_security/documents/mobile_forensics/Reference%20Mat-final-a.pdf.

[29] Joffray, Olivier. "Method and system for obtaining a PIN validation signal in a data processing unit", 2010.

[30] de Jong, Eduard. "Method and apparatus for protecting against side channel attacks against personal identification numbers", 2009.

[31] Joye, M. and F. Olivier. "Side-channel analysis". *Encyclopedia of Cryptography and Security*, 571–576, 2005.

[32] Kelsey, John, Bruce Schneier, David Wagner, and Chris Hall. "Side channel cryptanalysis of product ciphers". Jean-Jacques Quisquater, Yves Deswarte, Catherine Meadows, and Dieter Gollmann (editors), *Computer Security - ESORICS 98*, volume 1485 of *Lecture Notes in Computer Science*, 97–110. Springer Berlin Heidelberg, 1998. ISBN 978-3-540-65004-1. URL http://dx.doi.org/10.1007/BFb0055858.

[33] Kocher, Paul, Joshua Jaffe, and Benjamin Jun. "Introduction to Differential Power Analysis", 1998. URL http://www.cryptography.com/public/pdf/DPATechInfo.pdf.

[34] Kocher, Paul, Joshua Jaffe, and Benjamin Jun. "Differential Power Analysis". Michael Wiener (editor), *Advances in Cryptology CRYPTO 99*, volume 1666 of *Lecture Notes in Computer Science*, 388–397. Springer Berlin Heidelberg, 1999. ISBN 978-3-540-66347-8. URL http://dx.doi.org/10.1007/3-540-48405-1_25.

[35] Kocher, Paul C. "Cryptanalysis of Diffie-Hellman, RSA, DSS, and Other Systems Using Timing Attacks (Extended Abstract)". *Advances in Cryptology, CRYPTO '95: 15th Annual International Cryptology Conference*, 27–31. Springer-Verlag, 1995.

[36] Kömmerling, Oliver and Markus G Kuhn. "Design principles for tamper-resistant smartcard processors". *Proceedings of the USENIX Workshop on Smartcard Technology on USENIX Workshop on Smartcard Technology*, 2–2. USENIX Association, 1999.

[37] Koschuch, Manuel, Joachim Lechner, Andreas Weitzer, Johann Großschädl, Alexander Szekely, Stefan Tillich, and Johannes Wolkerstorfer. "Hardware/software co-design of elliptic curve cryptography on an 8051 microcontroller". *Cryptographic Hardware and Embedded Systems-CHES 2006*, 430–444. Springer, 2006.

[38] Le, Thanh-Ha, Cécile Canovas, and Jessy Clédière. "An overview of side channel analysis attacks". *Proceedings of the 2008 ACM symposium on Information, computer and communications security*, ASIACCS '08, 33–43. ACM, New York, NY, USA, 2008. ISBN 978-1-59593-979-1. URL http://doi.acm.org/10.1145/1368310.1368319.

[39] Mangard, Stefan, Elisabeth Oswald, and Thomas Popp. *Power analysis attacks: Revealing the secrets of smart cards*. Springer Science+ Business Media LLC, New York, NY, 2007.

[40] Messerges, Thomas S, Ezzy A Dabbish, and Robert H Sloan. "Investigations of power analysis attacks on smartcards". *USENIX workshop on Smartcard Technology*, volume 1999. 1999.

[41] Novak, Roman. "Side-channel attack on substitution blocks". *Applied Cryptography and Network Security*, 307–318. Springer, 2003.

[42] Oracle Corporation. "Java Card Specification 2.2.2", 2006. URL http://www.oracle.com/technetwork/java/javacard/specs-138637.html.

[43] Ortiz, C. Enrique. "An Introduction to Java Card Technology - Part 1", May 2003. URL http://www.oracle.com/technetwork/java/javacard/javacard1-139251.html.

[44] Peeters, Eric, François-Xavier Standaert, and Jean-Jacques Quisquater. "Power and electromagnetic analysis: Improved model, consequences and comparisons". *Integration, the VLSI journal*, 40(1):52–60, 2007.

[45] Quisquater, Jean-Jacques and David Samyde. "A new tool for non-intrusive analysis of smart cards based on electro-magnetic emissions: the SEMA and DEMA methods". *Eurocrypt rump session*, 2000.

[46] Rankl, Wolfgang and Wolfgang Effing. *Smart card handbook*. John Wiley & Sons, Ltd, fourth edition, 2010.

[47] Rao, Josyula R, Pankaj Rohatgi, Helmut Scherzer, and Stephane Tinguely. "Partitioning attacks: or how to rapidly clone some GSM cards". *Security and Privacy, 2002. Proceedings. 2002 IEEE Symposium on*, 31–41. IEEE, 2002.

[48] Schindler, Werner. "A Timing Attack against RSA with the Chinese Remainder Theorem". etinK. Ko and Christof Paar (editors), *Cryptographic Hardware and Embedded Systems CHES 2000*, volume 1965 of *Lecture Notes in Computer Science*, 109–124. Springer Berlin Heidelberg, 2000. ISBN 978-3-540-41455-1. URL http://dx.doi.org/10.1007/3-540-44499-8_8.

[49] STMicroelectronics. "Smartcard Security: Essential and Assurable!", 1998. URL http://www.st.com/stonline/press/news/year1998/t138ma.htm.

[50] Third-Generation Partnership Project. *3GPP TS 31.102 V5.14.0. Characteristics of the Universal Subscriber Identity Module (USIM) application*. 3GPP, 2005.

[51] Willassen, Svein. "Forensics and the GSM mobile telephone system". URL http://www.utica.edu/academic/institutes/ecii/publications/articles/A0658858-BFF6-C537-7CF86A78D6DE746D.pdf.

[52] Wilson, Tim. "Researcher Cracks Security Of Widely Used Computer Chip", Feb 2010. URL http://www.darkreading.com/security/news/222600843.

[53] Witteman, Marc. "Advances in smartcard security". *Information Security Bulletin*, 7(2002):11–22, 2002.

[54] van Woudenberg, Jasper GJ, Marc F Witteman, and Bram Bakker. "Improving differential power analysis by elastic alignment". *Topics in Cryptology–CT-RSA 2011*, 104–119. Springer, 2011.

[55] Zhou, Yuanyuan, Yu Yu, FX Standaert, and JJ Quisquater. "On the Need of Physical Security for Small Embedded Devices: a Case Study with COMP128-1 Implementations in SIM Cards".

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
13–06–2013	Master's Thesis	Oct 2011–Jun 2013

4. TITLE AND SUBTITLE

Side-channel Analysis of Subscriber Identity Modules

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Hearle, John Andrew

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB, OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-13-J-03

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

INTENTIONALLY LEFT BLANK

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Subscriber identity modules (SIMs) contain useful forensic data but are often locked with a PIN code that restricts access to this data. If an invalid PIN is entered several times, the card locks and may even destroy its stored data. This presents a challenge to the retrieval of data from the SIM when the PIN is unknown.

The field of side-channel analysis (SCA) collects, identifies, and processes information leaked via inadvertent channels. One promising side-channel leakage is that of electromagnetic (EM) emanations; by monitoring the SIM's emissions, it may be possible to determine the correct PIN to unlock the card.

This thesis uses EM SCA techniques to attempt to discover the SIM card's PIN. The tested SIM is subjected to simple and differential electromagnetic analysis. No clear data dependency or correlation is apparent. The SIM does reveal information pertaining to its validation routine, but the value of the card's stored PIN does not appear to leak via EM emissions. Two factors contributing to this result are the black-box nature of PIN validation and the hardware and software SCA countermeasures. Further experimentation on SIMs with known operational characteristics is recommended to determine the viability of future SCA attacks on these devices.

15. SUBJECT TERMS

Side channel analysis, subscriber identity modules

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Rusty O. Baldwin
U	U	U	UU	84	19b. TELEPHONE NUMBER *(include area code)* (937) 255-6565 x4445, rusty.baldwin@afit.edu